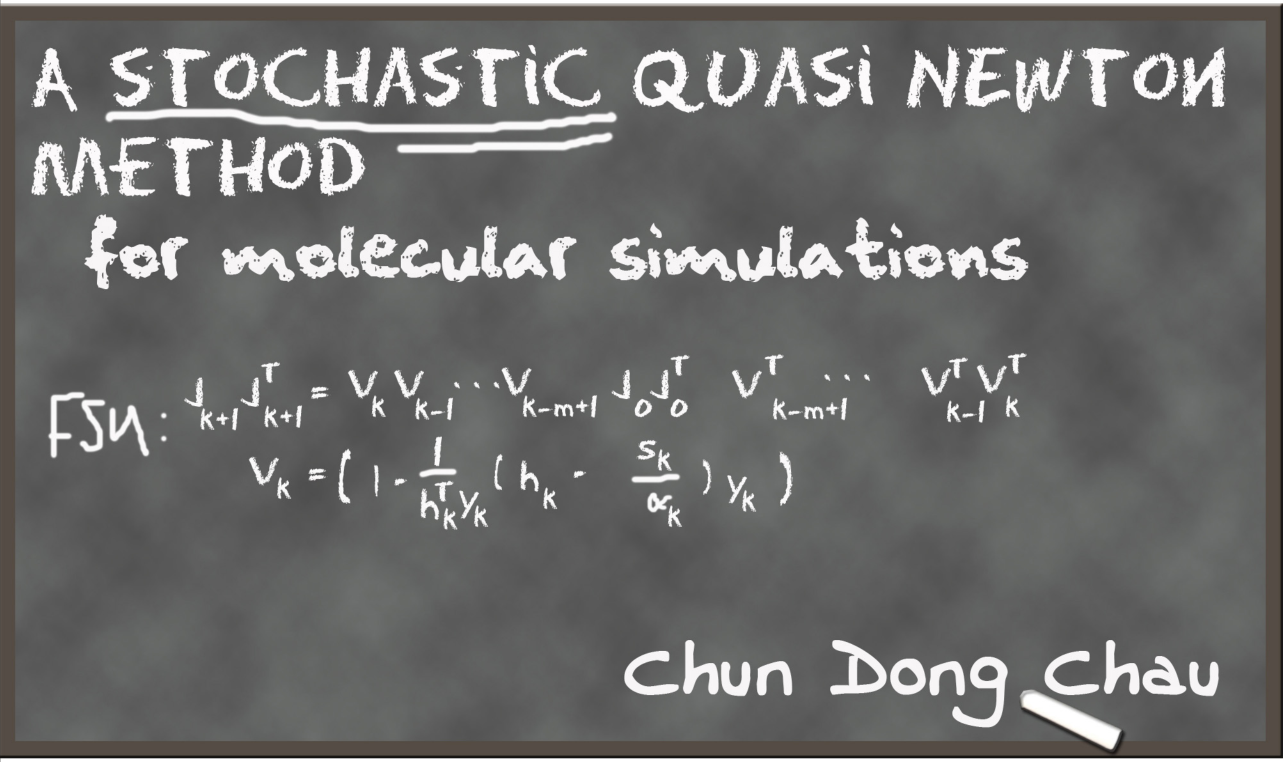


A STOCHASTIC QUASI NEWTON METHOD

for molecular simulations

FSU: $J_{k+1}J_{k+1}^T = V_k V_{k-1} \cdots V_{k-m+1} J_0 J_0^T V_{k-m+1}^T \cdots V_{k-1}^T V_k^T$

$$V_k = \left(I - \frac{1}{h_k^T y_k} \left(h_k - \frac{s_k}{\alpha_k} \right) y_k \right)$$

Chun Dong Chau

A Stochastic Quasi Newton Method for molecular simulations

Chun Dong Chau

A Stochastic quasi Newton method for molecular simulations

Chun Dong Chau

A Stochastic quasi Newton method for molecular simulations
C. D. Chau

Keywords: Stochastic Optimisation; Fokker Planck Equation; Quasi Newton; Langevin Dynamics; Limited Memory method.

Typeset in LaTeX.

A Stochastic quasi Newton method for molecular simulations

PROEFSCHRIFT

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van Rector Magnificus prof. mr. P. F. van der Heijden,
volgens besluit van het College voor Promoties
te verdedigen op woensdag 3 november 2010
klokke 15.00 uur

door

Chun Dong Chau

wiskundig ingenieur
geboren te Vlaardingen

in 1980

Promotiecommissie

promotor : Prof. dr. ir. J.G.E.M. Fraaije
co-promotores : Dr. G. J. A. Sevink
referent : Dr. E. M. Blokhuis
overige leden : Prof. dr. J. Brouwer
Prof. dr. W. J. Briels
Prof. dr. ir. F. A. M. Leermakers
Prof. dr. ir. S. de Leeuw
Prof. dr. F. Müller-Plathe
Prof. dr. P. Bolhuis
Prof. dr. ir. P.P.A.M. van der Schoot

"Two things are infinite: the universe and human stupidity;
and I’m not sure about the the universe."
Albert Einstein

To my family and the ones qualifying as family
because of our common denominator

Contents

CHAPTER 1

Introduction

A true challenge in simulating structural evolution in materials is the long time and large length scales that one needs to capture while accurately including important (atomic) details. As an example, polymers in melts, blends and solutions include structural features that span the whole range from nanometers, microns, millimeters to even larger sizes, which, depending on the property/phenomena of interest, may be important for particular material properties. The corresponding time scales of the dynamic process relevant for these different material properties vary from femtoseconds to milliseconds or even seconds or hours. This wide range of length and time scales is responsible for the variability of physical properties, making (rational) design of structures for intricate applications possible. Unfortunately there is no model or simulation algorithm that spans the whole range of length and time scales. For this reason, multiscale modeling - bridging and linking different computational methods that each provides a realistic description on well-defined time and length scales - is gaining importance for predicting and designing structure resulting in specific material properties.

The leading thread in this thesis is a stochastic Quasi-Newton (S-QN) model that contains both mathematical (sampling) and physical (diffusion) elements. In particular, we start from a conventional Langevin equation for particles diffusion, that uses a separation in time and length scales of different components of the system, for instance a polymer in solvent, to represent the fastest motions of the system (solvent) by a random fluctuation force. By averaging over the fastest motions, only the slower modes of the system are explicitly represented and the functional collective modes would hypothetically come within simulation reach. However, the resulting description in terms of coarse grained variables can also be categorized in terms of slow or fast modes, but now on a coarse-grained scale. Again, the fastest modes restrict the time step for stable time integration, resulting in substantial computational effort for capturing slow, collective motions on this scale. Since the sampling performance is always dictated by the smallest time or length scale in system descriptions, we seek to improve the simulation performance by automatically updating all modes roughly equally fast. Our method, described in this thesis, automatically accelerates the slow modes of the system. Since fast modes are simultaneously slowed down, we effectively integrate the Langevin equation with a much larger time step. The mode-scaling is caused by the proposed mobility and gives rise to much more efficient sampling of the energy landscape.

In Chapter 2 the new method is introduced: an alternative to the standard Langevin equations for Brownian dynamics. The focus is on introducing the new stochastic quasi Newton technique, which could both be seen as a quasi-Newton method with noise and Langevin equations with alternative mobility. We compare its performance to that of the standard technique and provide a detailed analysis of its thermodynamic and computational properties. We conclude that, even for lower dimensional systems, the new method is an order of magnitude more efficient in crossing energy barriers than the standard approach.

In Chapter 3, a new update scheme is introduced for updating the mobility matrix. The bottle-neck of our proposed method in Chapter 2 is the calculation of the mobility and its factorized form. Especially for the latter, an efficient update scheme in factorized form is needed for high-dimensional systems. Without such extension, practical application to molecular systems will be limited; with the extension, a new field is opened. Here, we focus on the details of such an algorithm and a detailed analysis of the performance and, in addition, discuss general guidelines for the choice of the free parameters in our methodology. We have chosen to focus on the physical context, i.e. critical slowing down phenomena and the automated separation of time scales in molecular simulation (multiplicity of timescales), and the conceptual relation to existing general methodology (Fourier acceleration techniques). Our results here are for a rather "academic" high-dimensional system but nevertheless relevant within this context. In particular, the system contains the critical slowing down features, represents a "stepping stone" for the simulation of more realistic systems and is very well suited for a quantitatively analysis of performance. The results are decisive in that the method is indeed capable of incorporating a multiplicity of time steps in an automated way.

Chapter 4 deals with an application of the new method to a minimal model of a protein. In order to avoid numerical problems, we consider the problem of ill-conditioning. This problem is solved by refining the update scheme for the mobility such that regularization occurs in the ill-conditioned cases. We analyze the effect of regularization and apply the new scheme to the minimal model of the chosen protein in a stepwise manner, by including an increasing number of terms of the total coarse-grained energy expression. A detailed analysis of the collective dynamics and sampling behavior for the considered coarse-grained protein is provided.

Although the proposed factorized update (FSU) in Chapter 3 makes the compute expensive factorization (Cholesky decomposition) redundant, it is useful to develop a limited memory implementation, much in the spirit of existing limited-memory optimization methods like L-BFGS. The proposed L-FSU method is not only more efficient in term of memory storage, since no full matrices need to be stored, but also computationally. Although the limited memory update for the new mobility was already introduced in Chapter 3, the incorporation within our S-QN method (especially the spurious drift term) is not straightforward. Chapter 5 provides detail how to integrate L-FSU and S-QN.

Although the proposed factorized update (FSU) in Chapter 3 makes the compute expensive factorization (Cholesky decomposition) redundant, it is useful to develop a limited memory implementation, much in the spirit of existing limited-memory optimization methods like L-BFGS. The proposed L-FSU method is not only more efficient in term of memory storage, since no full matrices need to be stored, but also computationally. Although the limited memory update for the new mobility was already introduced in Chapter 3, the incorporation within our S-QN method (especially the spurious drift term) is not straightforward. Chapter 5 provides detail how to integrate L-FSU and S-QN. Since our method is based on the Langevin equation, this introduction starts with the history of Brownian motion with its forthcoming Langevin equation. The Langevin equation consist of a deterministic part and stochastic/noise part, where the deterministic part only (NewtonŠs second law) gives the common approach in performing molecular simulations: molecular dynamics (MD). In addition, a purely stochastic approach is also available for performing molecular simulations: Monte Carlo (MC). Both simulation methods are shortly covered here. The new proposed mobility in the Langevin equation is a space dependent mobility. Since it is a space dependent mobility, it needs to be updated at every time step. The update scheme for the mobility is very much related to the update scheme in mathematical minimization for the quasi newton method. This method is briefly described in this chapter of preliminaries.

Last but not least in this chapter, an example of a system with a slow and fast mode is given and shows illustratively that integration with larger time steps for slow modes causes faster decrease towards (energy) minimum.

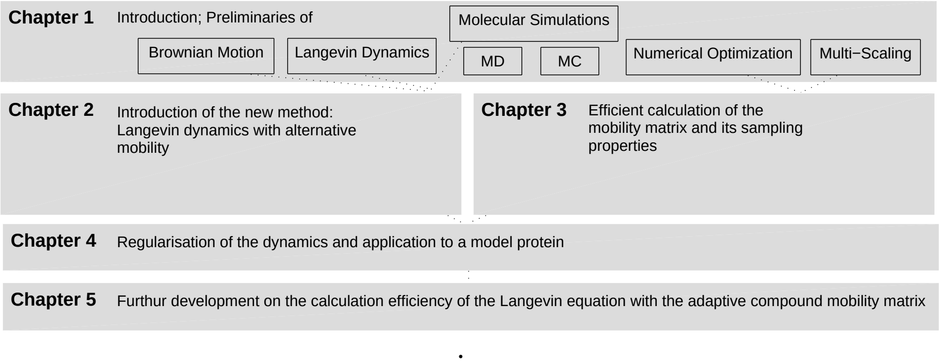


Figure 1.1: *Schematic drawing outline thesis*

1.1. Brownian Motion

Brownian motion is named after the Scottish botanist Robert Brown (1773 - 1858), who observed pollen grains with ’rapid oscillatory motion’ in water. However, it was not Brown who discovered this type of motion. In [1], the author cleverly point out that anyone looking at water through a microscope is able to see small things moving around. Brown himself also mentions precursors in his papers [2, 3]. Brown’s original contribution was that the motion is present in organic as well as inorganic matter (by that time one thought that the particles were alive), and that the motions have their origin in the particles themselves and not in the surrounding fluid. Unfortunately he couldn't work out what was causing the motion.

In 1905 Albert Einstein (1879- 1955) explained, in addition to his work on the special theory of relativity and the photoelectric effect, the Brownian motion. Einstein predicted that the random motions of molecules in a liquid impacting on larger suspended particles would result in irregular, random motions of the particles: the Brownian motion. He published his theory on Brownian motion in 1905 in a paper titled: 'On the movement of small particles in stationary liquids required by the molecular-kinetic theory of heat' [4]. This work was extended in his doctoral dissertation − also published in 1905 − that shows how to calculate Avogadro's number and hence the sizes of molecules [5].

Einstein derived the diffusion equation

$$\frac{\partial p}{\partial t} = D\nabla^2 p, \tag{1.1}$$

where $p = p(x, t)$ is the probability that a brownian particle is at x at time t and D is the so called diffusion constant. The solution (with initial condition $p(x, 0) = \delta(x)$, the delta distribution) is given as

$$p(x,t) = \frac{1}{(4\pi Dt)^{3/2}} \exp\left(-\frac{|x|^2}{4Dt}\right), \tag{1.2}$$

where $|\cdot|$ is the Euclidean norm for the multidimensional case . By considering a system of brownian particles in equilibrium between osmotic pressure and friction force which obeys Stokes' law, Einstein derived an expression for the diffusion coefficient. The expression is given in terms of temperature T, viscosity η and radius a of the Brownian particles

$$D = \frac{kT}{6\pi\eta a}, \tag{1.3}$$

where $k = k_B$ is the Boltzmann constant. By observations and equation(1.2) one can determine D and from (1.3) the Boltzmann constant is obtained.

Einstein argued that the velocity of a brownian particle can not be measured due to the rapid change of direction and magnitude of the particles in such a short time scale. He introduced the mean squared displacement $\overline{\Delta_x^2}$ of the displacement Δ_x in the x-direction in a certain time interval τ, of the Brownian particle as the observable quantity in Brownian motion. From the solution of the diffusion equation, the mean squared displacement could derived and is given by

$$\overline{\Delta_x^2} = \frac{RT}{N_{\text{av}}} \frac{1}{3\pi\eta a} \tau, \tag{1.4}$$

where R is the gas constant and N_{av} is the number of Avogadro.
In 1906, Marian Smoluchowski (1972 - 1917) derived independently from Einstein the same form for the mean squared displacement of a Brownian particle [6]. Smoluchowski's used a dynamical approach for describing the Brownian motion and obtained the mean squared displacement which differs from Einsteins by a factor $\frac{64}{27}$.
Paul Langevin (1872-1946), was well aware of the findings of Einstein and Smoluchowski regarding Brownian motion. Also using a dynamical approach Langevin was able to derive exact the same expression as Einstein for the mean squared displacement [7]. Langevin described the motion of free particles of mass m in direction x with two force terms: an according to Stokes' formula a viscous resistance term and an independent fluctuation force term X

$$m\frac{\mathrm{d}^2x}{\mathrm{d}t} = -6\pi\eta a\frac{\mathrm{d}x}{\mathrm{d}t} + X. \tag{1.5}$$

Multiplying equation(1.5) by x Langevin obtained

$$\frac{m}{2}\frac{\mathrm{d}^2x^2}{\mathrm{d}t} - mv^2 = -3\pi\eta a\frac{\mathrm{d}x^2}{\mathrm{d}t} + Xx. \tag{1.6}$$

Considering a large number of identical particles of mass m, making use of $m\overline{v^2} = RT/N_{\mathrm{av}}$ and averaging over equation(1.6) gives

$$\frac{m}{2}\frac{\mathrm{d}z}{\mathrm{d}t} + 3\pi\eta az = \frac{RT}{N_{av}}, \tag{1.7}$$

where $z = \overline{\frac{\mathrm{d}x^2}{\mathrm{d}t}}$. Notice that the average of Xx is zero due to the irregularity of X. Langevin found a solution for z which is in the regime of a constant rate of agitation given as

$$\frac{\overline{\mathrm{d}x^2}}{\mathrm{d}t} = \frac{RT}{N_{av}}\frac{1}{3\pi\mu a}. \tag{1.8}$$

In a time interval τ holds

$$\overline{x^2} - \overline{x_0^2} = \frac{RT}{N_{av}}\frac{1}{3\pi\mu a}\tau, \tag{1.9}$$

which is exact the same as Einsteins equation(1.4).
The validity of the expression for the displacement has been verified experimentally by Jean Perrin (1870-1942) and in 1908 he indeed confirmed that Einsteins expression for the displacement was exact [8].

Langevin's equation, a differential equation with a random force term X, was the first stochastic differential equation. Stochastic differential equations are widely used these days. For example in financial modeling (the famous Black-Scholes formula), neuroscience and off course as stochastic dynamics in molecular physics.
Throughout this thesis the following form of the Langevin equation will be considered

$$\boxed{\mathrm{d}\mathbf{x} = [-M(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T\nabla\cdot M(\mathbf{x})]\,\mathrm{d}t + \sqrt{2k_B T}L(\mathbf{x})\mathrm{d}W(t)}\,. \tag{1.10}$$

Below we show how this form of the Langevin equation is obtained and also discuss some basic properties of the stochastic differential equation.

1.2. Classical Langevin Equation and Stochastic Differential Equation

The most simple Langevin equation is the description of the Brownian motion

$$m\frac{\mathrm{d}v(t)}{\mathrm{d}t} = -\gamma v(t) + X(t), \tag{1.11}$$

where γ/m is the friction constant. Equation(1.11) can be written in a more general form of a stochastic differential equation (SDE)

$$\frac{\mathrm{d}x}{\mathrm{d}t} = a(x,t) + b(x,t)\zeta(t), \tag{1.12}$$

where x is the variable of interest, $a(x,t)$ and $b(x,t)$ are known functions and the fluctuation force $\zeta(t)$ is assumed to be a gaussian process with idealized mathematical description, i.e. for $t \neq t'$

$$\langle\zeta(t)\zeta(t')\rangle = \delta(t-t') \quad \text{and} \quad \langle\zeta(t)\rangle = 0. \tag{1.13}$$

Due to the idealization, the variable of interest x becomes a Markov process and therefore its transition probability is described by the Fokker-Planck equation (FPE)

$$\partial_t p(x,t|x_0,t_0) = -\partial_x[a(x,t)p(x,t|x_0,t_0)] + \frac{1}{2}\partial_x^2[b(x,t)^2 p(x,t|x_0,t_0)]. \tag{1.14}$$

By assuming the Brownian motion taken place in a medium in thermal equilibrium, we have

$$p_s(x) = \mathcal{N}\exp\left(-\frac{1}{2}\frac{mx^2}{kT}\right), \tag{1.15}$$

since the variable of interest x corresponds with the velocity v and therefore the stationary solution of the FPE must coincide with the Maxwell - Boltzmann distribution. Making use of $a(x,t) = -\frac{\gamma}{m}v = a$, one can obain $b(x,t) = b$ from

$$\frac{2a}{b^2} = p_s \partial_x p_s. \tag{1.16}$$

$$\Longrightarrow$$

$$p_s = C \exp\left(\int \frac{2a}{b^2}\,\mathrm{d}x\right). \tag{1.17}$$

Calculating the integral

$$\int \frac{2a}{b^2}\,\mathrm{d}x = -\int \frac{2\gamma v}{m}\frac{1}{b^2}\,\mathrm{d}v = -\frac{\gamma v^2}{mb^2}, \tag{1.18}$$

and setting it equal to $-\frac{1}{2}\frac{mv^2}{kT}$ yields

$$b^2 = \frac{2\gamma kT}{m^2}, \tag{1.19}$$

which is the relation between the systematic (frictional) force $a(v,t) = -\frac{\gamma}{m}v(t)$ and the random force $b(v,t)\zeta(t) = \frac{X(t)}{m}$: the so called *fluctuation dissipation theorem,*

$$\frac{2\gamma kT}{m^2}\langle\zeta(t)\zeta(t')\rangle = \frac{1}{m^2}\langle X(t)X(t')\rangle \tag{1.20}$$

$$\Longrightarrow$$

$$\langle X(t)X(t')\rangle = 2\gamma kT\delta(t-t'). \tag{1.21}$$

Similar derivation and results hold for Brownian motion applied in a potential field Φ, where the Langevin equation becomes

$$m\frac{\mathrm{d}v(t)}{\mathrm{d}t} = -\gamma v(t) - \partial_x\Phi(x) + X(t), \tag{1.22}$$

and where the considered stationary solution for the transition probability equals

$$p_s(x,v) = \mathcal{N}\exp\left(-\frac{1}{kT}\{\tfrac{1}{2}mv^2 + \Phi(x)\}\right). \tag{1.23}$$

In this thesis we will consider particles in over-damped limit. Which means that the Brownian behavior of particles interacting through a potential is in the high friction limit. Under this assumption one finds the positional Langevin equation ($\frac{\mathrm{d}v}{\mathrm{d}t} = 0$)

$$\gamma v(t) = -\partial_x\Phi(x) + X(t). \tag{1.24}$$

The corresponding integral equation of (1.12) is given as

$$x(t) - x(0) = \int_0^t a[x(s), s]\,\mathrm{d}s + \int_0^t b[x(s), s]\zeta(s)\,\mathrm{d}s. \tag{1.25}$$

Assuming $\int_0^t \zeta(s)\,\mathrm{d}s = W(t)$, one can write

$$\int_0^t b[x(s), s]\zeta(s)\,\mathrm{d}s = \int_0^t b[x(s), s]\,\mathrm{d}W(s), \tag{1.26}$$

which is a stochastic integral with respect to a sample function $W(s)$. More details on stochastic integrals can be found in [9]. Here, we only consider the integrated form of the positional Langevin equation

$$\mathrm{d}x = -\gamma^{-1}\partial_x\Phi(x)\,\mathrm{d}t + \sqrt{2\gamma^{-1}kT}\,\mathrm{d}W(t). \tag{1.27}$$

From the assumption $\int_0^t \zeta(s)\,\mathrm{d}s = W(t)$, one finds that $W(t)$ is a markov process with its corresponding FPE given by (1.14), with $a(x,t) = 0$ and $b(x,t) = 1$. By solving the Fokker Planck equation, with the delta distribution as its initial condition, $W(t)$ is determined as a Gaussian process with $\langle W(t)\rangle = W(t_0) = w_0$ and $\langle [W(t) - w_0]^2\rangle = t - t_0$, which will be called the Wiener process. More precisely the corresponding FPE is of the same form as Einsteins diffusion equation (1.1), therefore this one dimensional Wiener process is also often called Brownian motion. Notice that due to the irregularity of the sample paths (brownian motion), $W(t)$ is not differentiable which contradicts that the integral of $\zeta(t)$ is $W(t)$. Nevertheless, equation (1.27) can be used and interpreted consistently.

The positional Langevin equation in the multi-dimensional form with space dependent $\gamma = \gamma(\mathbf{x})$ is derived in the same way and given as

$$\mathrm{d}\mathbf{x} = [-M(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T\nabla\cdot M(\mathbf{x})]\,\mathrm{d}t + \sqrt{2k_B T}L(\mathbf{x})\mathrm{d}W(t), \tag{1.28}$$

where $M(\mathbf{x}) = \gamma(\mathbf{x})^{-1}$ is the mobility, $M(\mathbf{x}) = L(\mathbf{x})L^T(\mathbf{x})$ and $W(t)$ becomes the multivariate Wiener process with $\langle \mathrm{d}W_i(t)\mathrm{d}W_j(t)\rangle = \delta_{ij}\mathrm{d}t$. More details of the derivation is found in appendix A.

1.3. Molecular simulation preliminaries

There are two main approaches for performing molecular simulations: a deterministic approach and a stochastic approach. The deterministic approach, called Molecular

Dynamics (MD), simulates the time evolution of the molecular system and gives the actual trajectory of the system. The stochastic approach, Monte Carlo (MC), explores the energy surface by randomly accessing different configurations of the molecular system. The different configurations allows the calculation of the system's thermodynamic properties. In the remainder of this section the most common molecular dynamic and Monte Carlo methods are given.

1.3.1. Molecular dynamics

In MD the classical equations of motion are solved in time for a set of atoms. Newton's equation of motion can be discretized by Taylor expansion

$$\mathbf{r}_i(t+\Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\Delta t^2}{2m_i}\mathbf{f}_i(t) + \frac{\Delta t^3}{3!}\dddot{\mathbf{r}}_i(t) + \mathcal{O}(\Delta t^4), \tag{1.29}$$

$$\mathbf{v}_i(t+\Delta t) = \mathbf{v}_i(t) + \mathbf{f}_i(t)\frac{\Delta t}{m_i} + \frac{\Delta t^2}{2}\ddot{\mathbf{v}}_i(t) + \frac{\Delta t^3}{3!}\dddot{\mathbf{v}}_i(t) + \mathcal{O}(\Delta t^4), \tag{1.30}$$

where $\mathbf{r}_i$ and $\mathbf{v}_i$ is the position and velocity respectively of particle i and $\mathbf{f}_i$ the force on particle i. The most simple scheme is the Euler scheme which calculates the positions using the first three terms of expansion (1.29) ($\mathcal{O}(\Delta t^2)$) and calculates the velocities using the first two terms of expansion (1.30) ($\mathcal{O}(\Delta t)$). Not only is the Euler algorithm an unstable scheme, it is neither time-reversible nor phase-space preserving, which are important properties in physical systems.

1.3.1a. Verlet

It is possible to solve Newton's equation of motion without explicitly calculating the velocity. Consider the Taylor expansion at $t - \Delta t$

$$\mathbf{r}_i(t-\Delta t) = \mathbf{r}_i(t) - \mathbf{v}_i(t)\Delta t + \frac{\Delta t^2}{2m_i}\mathbf{f}_i(t) - \frac{\Delta t^3}{3!}\dddot{\mathbf{r}}_i(t) + \mathcal{O}(\Delta t^4). \tag{1.31}$$

The Verlet scheme [10, 11] for updating the positions is obtained by adding equations (1.29) and (1.31), which gives after simple algebraic operations

$$\mathbf{r}_i(t+\Delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t-\Delta t) + \frac{\Delta t^2}{m_i}\mathbf{f}_i(t) + \mathcal{O}(\Delta t^4), \tag{1.32}$$

and analogously by substracting the velocity is obtained

$$\mathbf{v}_i(t) = \frac{\mathbf{r}_i(t+\Delta t) - \mathbf{r}_i(t-\Delta t)}{2\Delta t} + \mathcal{O}(\Delta t^2). \tag{1.33}$$

Although the velocity is not necessary needed for determining the positions in the Verlet scheme, they are often needed to calculate physical quantities like the kinetic energy. However the update scheme for the velocity in the basic Verlet algorithm needs the positions at $t + \Delta t$ to calculate the velocity at time t. This deficiency can be solved by using the Leap Frog scheme, where the velocities are updated at half time steps and 'leap' ahead the positions [12].

1.3.1b. Velocity Verlet

An better implementation of the same basic algorithm is the so-called velocity Verlet scheme, where positions, velocities and accelerations at time $t + \Delta t$ are obtained from the same quantities at time t in the following way:

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t + \frac{\Delta t^2}{2m_i}\mathbf{f}_i(t) && (1.34) \\ \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t + \Delta t/2) + \frac{\Delta t}{2m_i}\mathbf{f}_i(t + \Delta t), && (1.35) \end{aligned}$$

where $\mathbf{v}_i(t + \Delta t/2)$ is given by

$$\mathbf{v}_i(t + \Delta t/2) = \mathbf{v}_i(t) + \frac{\Delta t}{2m_i}\mathbf{f}_i(t). \qquad (1.36)$$

The velocity Verlet scheme is very stable and is a widely used MD algorithm [13].

1.3.2. Monte Carlo methods

The Monte Carlo method is the most popular method used to generate a certain assemble. Assume that we want to calculate the integral

$$I = \int_a^b f(x)\,\mathrm{d}x, \qquad (1.37)$$

which can be rewritten as

$$I = (b - a)\langle f(x)\rangle \approx (b - a)\frac{1}{L}\sum_{i=1}^{L} f(x_i), \qquad (1.38)$$

where L is the number of random (hence, the name Monte Carlo) points x_i taken between (a, b). The variance of $\langle f(x)\rangle$ equals

$$\sigma^2 = \frac{1}{L}[\langle f(x)^2\rangle - \langle f(x)\rangle^2]. \qquad (1.39)$$

Thus for sufficient large L the discrete approximation of the integral is quite accurate. Clearly, the method works best for constant f. Consider $a = 0$ and $b = 1$ and introduce the weight function $w(x)$

$$I = \int_0^1 w(x)\frac{f(x)}{w(x)}\,\mathrm{d}x = \int_0^1 \frac{f(x(y))}{w(x(y))}\,\mathrm{d}y, \tag{1.40}$$

where $w(x)$ satisfies

$$\int w(x)\,\mathrm{d}x = 1 \quad y(x) = \int_0^x w(x')\,\mathrm{d}x',$$

such that $\mathrm{d}y = w(x)\mathrm{d}x$, $y(0) = 0$ and $y(1) = 1$ hold. For making use of Monte Carlo sampling for $\frac{f(x(y))}{w(x(y))}$ with random sampling in y from the distribution $w(x)$, $w(x)$ must be chosen such that the fraction is nearly constant

$$I = \frac{1}{L}\sum_{i=1}^{L}\frac{f(x(y_i))}{w(x(y_i))}. \tag{1.41}$$

Since the inversion of $y(x)$ is almost always difficult to obtain, one needs another method to approach the integral I.

1.3.3. Importance Sampling

Consider a canonical ensemble where the corresponding integral is denoted by

$$\langle A\rangle_{NVT} = \int p(\mathbf{x})A(\mathbf{x})\,\mathrm{d}\mathbf{x} = \langle\frac{A(\mathbf{x})p(\mathbf{x})}{w(\mathbf{x})}\rangle_w, \tag{1.42}$$

where $\langle \quad \rangle_w$ represents an average over all samples of $\mathbf{x}$ from the distribution $w(\mathbf{x})$, see equation (1.40) where $f(x)$ is replaced by $A(\mathbf{x})p(\mathbf{x})$. If $w(\mathbf{x}) = p(\mathbf{x})$ then $\langle A\rangle_{NVT} = \langle A\rangle_w$. The problem is how to take the samples such that at the end, $\mathbf{x}$ is distributed according to the chosen distribution. Before introducing the method found by Metropolis *et al* [14], consider first the following.

Consider a large number of particles moving independently through the space. Define $\rho_n(X)$ as the density of particles at X after n-movements. Then the net number of particles moving from X to Y in the next step equals

$$\begin{aligned}\Delta N &= \rho_n(X)\pi(X\to Y) - \rho_n(Y)\pi(Y\to X)\\ &= \rho_n(Y)\pi(X\to Y)\left[\frac{\rho_n(X)}{\rho_n(Y)} - \frac{\pi(Y\to X)}{\pi(X\to Y)}\right],\end{aligned} \tag{1.43}$$

where $\pi(X \to Y)$ is the probability that the particle will move from X to Y. If there are lots of particle in X then $\Delta N(X) > 0$, hence particles are moving from X to Y (towards the equilibrium state). The expression in the square brackets in (1.43) also shows that equilibrium is obtained if

$$\frac{\rho_n(X)}{\rho_n(Y)} = \frac{\pi(Y \to X)}{\pi(X \to Y)} = \frac{\rho_e(X)}{\rho_e(Y)}, \tag{1.44}$$

where $\rho_e(X)$ is the density of particles at X in equilibrium state. From here we go back to the problem how to take samples such that the samples form the distribution $w(X)$. We need to couple $\rho_e(X)$ and $w(X)$. The transition probability $\pi(X \to Y)$ can be rewritten as

$$\pi(X \to Y) = \alpha(X \to Y)\kappa(X \to Y), \tag{1.45}$$

where $\alpha(X \to Y)$ is probability of making a trial step from $X \to Y$ and $\kappa(X \to Y)$ is the probability of accepting this step. In equilibrium the following holds

$$\frac{\rho_e(X)}{\rho_e(Y)} = \frac{\pi(Y \to X)}{\pi(X \to Y)} = \frac{\alpha(Y \to X)\kappa(Y \to X)}{\alpha(X \to Y)\kappa(X \to Y)} = \frac{\kappa(Y \to X)}{\kappa(X \to Y)}, \tag{1.46}$$

where the latter equality holds if we assume that the trial step probability from X to Y is the same as the trial step probability back (from Y to X), i.e. $\alpha(X \to Y) = \alpha(Y \to X)$. Rest us to define the acceptance probability, which gives us the chance to include the distribution function $w(X)$. Metroplis *et al* introduced the following asymmetric scheme by defining the acceptance probability in the following way

$$\left.\begin{aligned} \kappa(Y \to X) &= 1 \\ \kappa(X \to Y) &= \frac{w(Y)}{w(X)} \end{aligned}\right\} \quad \text{if } w(X) > w(Y), \tag{1.47}$$

and in the other case

$$\left.\begin{aligned} \kappa(X \to Y) &= 1 \\ \kappa(Y \to X) &= \frac{w(X)}{w(Y)} \end{aligned}\right\} \quad \text{if } w(X) < w(Y). \tag{1.48}$$

Hence, in both the cases the equilibrium state satisfies

$$\frac{\rho_e(X)}{\rho_e(Y)} = \frac{w(X)}{w(Y)}. \tag{1.49}$$

The transition probability becomes

$$\pi(X \to Y) = \begin{cases} \alpha(X \to Y) & w(Y) \geq w(X) \quad (1.50a) \\ \\ \alpha(X \to Y)\frac{w(Y)}{w(X)} & w(Y) < w(X) \quad (1.50b) \end{cases}$$

$$\pi(X \to X) = 1 - \sum_{X \neq Y} \pi(X \to Y). \tag{1.51}$$

This gives the following algorithm for drawing samples from distribution $w(\mathbf{x})$

Algorithm 1.3.1 *Importance Sampling Monte Carlo*

1. *initiate* $\mathbf{x}_n$ *at random*
2. *change* $\mathbf{x}_n$ *with a random* $\Delta\mathbf{x}$
3. *calculate* $\frac{w(\mathbf{x}_n+\Delta\mathbf{x})}{w(\mathbf{x}_n)}$
4. *draw a random number r between* $(0, 1)$
    - *if* $r < \min(1, \frac{w(\mathbf{x}_n+\Delta\mathbf{x})}{w(\mathbf{x}_n)})$
      then $\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta\mathbf{x}$
    - *else* $\mathbf{x}_{n+1} = \mathbf{x}_n$
5. *repeat from step 2 for* $n = n + 1$

1.3.3a. Force-Bias Monte Carlo

It is not necessary that $\alpha(X \to Y) = \alpha(Y \to X)$. If $\alpha(X \to Y) \neq \alpha(Y \to X)$, then the acceptance probability needs to be modified without violating $\frac{\rho_e(X)}{\rho_e(Y)} = \frac{w(X)}{w(Y)}$. The acceptance probability becomes

$$\left.\begin{aligned} \kappa(Y \to X) &= 1 \\ \kappa(X \to Y) &= \frac{\alpha(Y \to X)}{\alpha(X \to Y)}\frac{w(Y)}{w(X)} \end{aligned}\right\} \quad \text{if } \alpha(X \to Y)w(X) > \alpha(Y \to X)w(Y), \tag{1.52}$$

and in the other case

$$\left.\begin{aligned} \kappa(X \to Y) &= 1 \\ \kappa(Y \to X) &= \frac{\alpha(X \to Y)}{\alpha(Y \to X)}\frac{w(X)}{w(Y)} \end{aligned}\right\} \quad \text{if } \alpha(X \to Y)w(X) < \alpha(Y \to X)w(Y). \tag{1.53}$$

The transition probability becomes

$$\pi(X \to Y) = \begin{cases} \alpha(X \to Y) & \alpha(X \to Y)w(X) \leq \alpha(Y \to X)w(Y) \quad (1.54a) \\ \alpha(X \to Y)\frac{\alpha(Y \to X)}{\alpha(X \to Y)}\frac{w(Y)}{w(X)} & \alpha(X \to Y)w(X) > \alpha(Y \to X)w(Y) \quad (1.54b) \end{cases}$$

$$\pi(X \to X) = 1 - \sum_{X \neq Y} \pi(X \to Y). \tag{1.55}$$

Step 4 in algorithm 1.3.1 should now be modified by changing

$$\min(1, \frac{w(\mathbf{x}_n + \Delta\mathbf{x})}{w(\mathbf{x}_n)}) \quad \text{into} \quad \min(1, \frac{\alpha(\mathbf{x}_n + \Delta\mathbf{x} \to \mathbf{x}_n)w(\mathbf{x}_n + \Delta\mathbf{x})}{\alpha(\mathbf{x}_n \to \mathbf{x}_n + \Delta\mathbf{x})w(\mathbf{x}_n)}). \tag{1.56}$$

The trial probability function α is included in the acceptance criteria of the algorithm. Taking a closer look to the definition of the trial probability, one can conclude that incorporating α in the criteria, gives us an opportunity to bias moves to preferred directions. Back to the canonical ensemble, with a system of N particles, we have

$$w(\mathbf{x}) = \mathcal{N} \exp[-\beta\Phi(\mathbf{x})], \tag{1.57}$$

where $\beta = \frac{1}{kT}$, $\Phi(\mathbf{x})$ is the potential energy and $\mathbf{x}$ is a vector of length $3N$. It is obvious that when the trial probability function is defined by $\alpha(\mathbf{x} + \Delta\mathbf{x} \to \mathbf{x}) = w(\mathbf{x})$ and $\alpha(\mathbf{x} \to \mathbf{x} + \Delta\mathbf{x}) = w(\mathbf{x} + \Delta\mathbf{x})$ that then every move will be accepted. Unfortunately it is not possible to do so, since this will require pre-knowledge of the whole distribution. For the canonical ensemble, we can approach the trial distribution α by expanding $w(\mathbf{x} + \Delta\mathbf{x})$ around $\mathbf{x}$, which means for one particle move

$$\alpha(x_i \to x_i + \Delta x_i) = c \exp[-\beta(\nabla_{x_i}\Phi(\mathbf{x}) \cdot \Delta x_i)], \tag{1.58}$$

where c is the normalization constant. Since $\Phi(\mathbf{x})$ is the potential energy, $-\nabla_{x_i}\Phi(\mathbf{x})$ is the force $f_i(\mathbf{x})$ on particle i in state $\mathbf{x}$. This shows that displacements in the direction of the force are more often accepted. From here we will use the abbreviation α_{on} for trial probability from the old state to the new state $\alpha(x_i \to x_i + \Delta x_i)$ and vice versa α_{no} for the trial probability $\alpha(x_i + \Delta x_i \to x_i)$. For $w(\mathbf{x})$ we will do the same, i.e. $w(\mathbf{x}) = w_o$ and $w(\mathbf{x} + \Delta x) = w_n$. The displacement is written as $\Delta x_i = x_i^n - x_i^o = \Delta x_i^{on}$ and the force on particle i in the old state is written as f_i^o and similarly in the new state as f_i^n. The acceptance ratio becomes

$$\begin{aligned} \frac{\alpha_{no} w_n}{\alpha_{on} w_o} &= \frac{c_n}{c_o} \frac{\exp[\beta f_i^n (x_i^o - x_i^n)]}{\exp[\beta f_i^o (x_i^n - x_i^o)]} \frac{\exp[-\beta \Phi(\mathbf{x} + \Delta \mathbf{x})]}{\exp[-\beta \Phi(\mathbf{x})]} \\ &= \frac{c_n}{c_o} \exp[-\beta((f_i^o + f_i^n) \Delta x_i^{on} + \Delta \Phi)], \end{aligned} \tag{1.59}$$

where $\Delta\Phi$ is the change in energy. This scheme is called the force-bias scheme by Pangali, Rao and Berne [15]. The displacement Δx_i^{on} here is a random displacement. In the next subsection we relate this displacement with Brownian motion.

1.3.3b. Smart Monte Carlo

P.J. Rossky, J.D. Doll and H.L. Friedman [16] used the displacement of the brownian dynamics from D.L. Ermak [17] as the trial displacement Δx_i^{on} of particle i from the old state to the new state

$$\Delta x_i^{on} = \beta A f_i^o + \delta x_i, \tag{1.60}$$

where δx_i is a Gaussian random variable, with zero mean and variance $2A$. The trial distribution function equals

$$\alpha^{on} = (4A\pi)^{-\frac{3}{2}} \exp[-(\Delta x_i^{no} - \beta A f_i^o)^2 \frac{1}{4A}], \tag{1.61}$$

and the acceptance ratio equals

$$\frac{\alpha_{no} w_n}{\alpha_{on} w_o} = \exp[-\beta(\Delta\Phi + \frac{(f_i^n + f_i^o)}{2} \Delta x_i^{no} + \frac{\beta A}{4}((f_i^n - f_i^o)^2 + 2 f_i^o (f_i^n - f_i^o)))]. \tag{1.62}$$

Notice the similarities between the force-bias and smart Monte Carlo method.

1.3.4. Simulated Annealing

Simulated annealing was found by modeling a physical phenomenon, and is a generalization of a Monte Carlo method for global optimization. In a physical annealing process, a melt is slowly cooled down. The initially high temperature of the melt corresponds to a disordered system and as cooling proceeds, the system becomes more ordered. Due to the slowly cooling down, the system is in thermodynamic equilibrium any time, and at the end (where the temperature approaches zero) the lowest energy state is obtained. If the initial temperature is too low or cooling is done too

fast, the system may become quenched, forming defects or freezing out in metastable states (i.e. trapped in a local minimum energy state).
At the initial high temperature Monte Carlo sampling is applied and after sufficient time the temperature is decremented and again a Monte Carlo sampling is applied. The entire process is repeated until a frozen state is achieved at T=0, corresponding with the lowest energy state and thus the global minimum of the energy potential Φ.

Simulated annealing is also applicable to combinatorial problem [18, 19]. The energy potential corresponds to the objective function and each state of the thermodynamic system correspond to the current solution of the combinatorial problem. The difficulty is finding an analogy in the combinatorial problem for the temperature. For more details we refer to [20].

1.4. Mathematical preliminaries on Unconstrained Optimization

Using the potential energy function to find the geometry of a molecule (or an assemblage of molecules) which corresponds to a minimum potential energy, is a very common goal in molecular simulations. The minimization of the potential energy function (i.e., geometry optimization) involves finding the minima of the potential field Φ. The minima correspond to stable states of the system; any movement away from a minimum gives a higher energy. Since finding a minimum is a typical problem in mathematics, it is not surprising that fundamental mathematical methods has been used in the search for these stable states.

1.4.1. Introduction in numerical mathematics

In unconstrained optimization an objective function $f(\mathbf{x})$, $f : \mathbb{R}^n \to \mathbb{R}$, has to be minimized with respect to $\mathbf{x}$. Mathematically, this can be written as

$$\min_{\mathbf{x}} f(\mathbf{x}). \tag{1.63}$$

Since $n \gg 1$ for large systems, one often needs a numerical approach to solve equation (1.63). There are two common approaches in numerical minimization: the line search strategy and the trust region strategy. Trust region methods first choose a step size and approximate the objective function within a certain region, then a direction is chosen for the displacement in this trust region. Line search methods first choose a direction and then a step size is chosen. Our method is partially based on the this method. Below we describe the most popular line search method: the quasi Newton (QN) method.

1.4.2. Quasi Newton Method

The line search method searches along a known direction **p** with unknown step length ϵ using

$$f(\mathbf{x}_k + \epsilon\mathbf{p}) = f_k + \epsilon\mathbf{p}^T\nabla f_k + \frac{1}{2}\epsilon^2\mathbf{p}^T\nabla^2 f(\mathbf{x}_k + \tau\mathbf{p})\mathbf{p}, \tag{1.64}$$

where $\tau \in (0, \epsilon)$ and $f_k = f(\mathbf{x}_k)$. The popular Newton direction is found by writing the object function as,

$$f(\mathbf{x}_k + \mathbf{p}) \approx f_k + \mathbf{p}^T\nabla f_k + \frac{1}{2}\mathbf{p}^T\nabla^2 f_k\mathbf{p} =: m_k(\mathbf{p}). \tag{1.65}$$

The approximation $f(\mathbf{x}_k + \mathbf{p}) \approx m_k(\mathbf{p})$ is very accurate for small $||\mathbf{p}||$. The unknown direction **p** is found by setting the derivative with respect to $\mathbf{p}_k$ of $m_k(\mathbf{p}_k)$ to zero. This gives the direction

$$\mathbf{p}_k = -\nabla^2 f_k^{-1}\nabla f_k, \tag{1.66}$$

which requires the existence of the inverse of $\nabla^2 f_k$. Therefore, $\nabla^2 f_k$ must be positive definite. If the inverse Hessian exists the direction (1.66) has the descending property, since for the second term of (1.64) holds

$$\nabla f_k^T\mathbf{p}_k = -\mathbf{p}_k^T\nabla^2 f_k\mathbf{p}_k \leq 0. \tag{1.67}$$

Not only the uncertainty of the existence of the inverse is a drawback in the Newton method but also it requires to calculate the Hessian $\nabla^2 f(\mathbf{x})$, which becomes a very expensive job for large systems. A method that avoids the calculation of the second derivatives, by using an iterative estimate of the Hessian, is called the Quasi Newton method.

In the next section the Quasi Newton method, BFGS (Broyden, Fletcher, Goldfarb and Shanno), will be described.

Using Taylor the following equation for a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ holds

$$f(\mathbf{x} + \mathbf{p}) = f(\mathbf{x}) + \nabla f(\mathbf{x} + \tau\mathbf{p}), \tag{1.68}$$

where $\tau \in (0, 1)$. Moreover, if f is twice continuously differentiable then the following holds

$$\begin{aligned} \nabla f(\mathbf{x}+\mathbf{p}) &= \nabla f(\mathbf{x}) + \int_0^1 \nabla^2 f(\mathbf{x}+\tau\mathbf{p})\mathbf{p}\,\mathrm{d}\tau \\ &= \nabla f(\mathbf{x}) + \nabla^2 f(\mathbf{x})\mathbf{p} + \int_0^1 [\nabla^2 f(\mathbf{x}+\tau\mathbf{p}) - \nabla^2 f(\mathbf{x})]\mathbf{p}\,\mathrm{d}\tau \end{aligned} \tag{1.69}$$

After setting $\mathbf{x} = \mathbf{x}_k$ and $\mathbf{p} = \mathbf{x}_{k+1} - \mathbf{x}_k$, one obtains

$$\nabla f_{k+1} = \nabla f_k + \nabla^2 f_{k+1}(\mathbf{x}_{k+1} - \mathbf{x}_k) + O(||\mathbf{x}_{k+1} - \mathbf{x}_k||),$$

which gives after neglecting the (small) last term on the right hand side of the equation

$$\underbrace{\nabla f_{k+1} - \nabla f_k}_{\mathbf{y}_k} \approx \underbrace{\nabla^2 f_{k+1}}_{H_{k+1}} \underbrace{(\mathbf{x}_{k+1} - \mathbf{x}_k)}_{\mathbf{s}_k}. \tag{1.70}$$

The equation with H_{k+1} as the Hessian approximation,

$$H_{k+1}\mathbf{s}_k = \mathbf{y}_k, \tag{1.71}$$

is called the secant equation, which guarantees that the gradient of $m_{k+1}(\mathbf{p})$,

$$\nabla[m_{k+1}(\mathbf{p})] = \nabla[f_{k+1} + \nabla f_{k+1}^T\mathbf{p} + \frac{1}{2}\mathbf{p}^T B_{k+1}\mathbf{p}], \tag{1.72}$$

matches the gradient of the objective function f at the latest two iterates $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$. The approximated Hessian needs to be positive definite, which holds if $\mathbf{s}_k$ and $\mathbf{y}_k$ satisfy the curvature condition

$$\mathbf{s}_k^T\mathbf{y}_k > 0. \tag{1.73}$$

When the object function f is strongly convex, the curvature condition is valid for any two points $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$. For nonconvex functions the curvature condition holds if the following conditions are set for the line search

$$f(\mathbf{x}_k + \epsilon_k\mathbf{p}_k) \leq f(\mathbf{x}_k) + c_1\epsilon_k\nabla f_k^T\mathbf{p}_k, \tag{1.74a}$$

$$\nabla f(\mathbf{x}_k + \epsilon_k\mathbf{p}_k)^T\mathbf{p}_k \geq c_2\nabla f_k^T\mathbf{p}_k, \tag{1.74b}$$

with $0 < c_1 < c_2 < 1$ and where ϵ_k is the step length in the direction $\mathbf{p}_k$. These conditions are called the Wolfe conditions which guarantees sufficient decrease in the object function. If the second condition holds then for condition (1.73) holds

$$\mathbf{s}_k^T \mathbf{y}_k \geq (c_2 - 1)\epsilon_k \nabla f_k^T \mathbf{p}_k, \tag{1.75}$$

where due to the descending direction of $\mathbf{p}_k$ the right hand side is positive and therefore, the curvature condition holds. Hence, the secant equation (1.71) has a solution. The secant equation gives n conditions. The requirement of positive definiteness of H_{k+1} gives also n conditions. But H_{k+1} has $\frac{n(n+1)}{2}$ degrees of freedom, so H_{k+1} is not uniquely determined. To determine H_{k+1} uniquely, an additional condition will be imposed; among all symmetric matrices, H is in some sense, closest to the current matrix H_k. This gives the following problem

$$\min_H \|H - H_k\|, \tag{1.76a}$$

subject to

$$H = H^T, \;\; H\mathbf{s}_k = \mathbf{y}_k, \;\; \text{and } \mathbf{v}^T H \mathbf{v} > 0 \text{ for all } \mathbf{v} \neq 0, \mathbf{v} \in \mathbb{R}^n. \tag{1.76b}$$

Different matrix norms can be used in (1.76a). A norm that allows easy solution of the minimization problem is the weighted Frobenius norm

$$\|A\|_W = \|W^{1/2} A W^{1/2}\|_F, \quad \text{with} \quad \|A\|_F = \Big(\sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}^2\Big)^{1/2}. \tag{1.77}$$

The weight matrix W is chosen such that the relation $W\mathbf{y}_k = \mathbf{s}_k$ is satisfied. The weight W is chosen as the inverse of the average Hessian defined by

$$[\overline{G}_k]^{-1} = \Big[\int_0^1 \nabla^2 f(\mathbf{x}_k + \tau\epsilon_k \mathbf{p}_k)\, \mathrm{d}\tau\Big]^{-1}, \tag{1.78}$$

having the property $\mathbf{y}_k = \overline{G}_k \epsilon_k \mathbf{p}_k = \overline{G}_k \mathbf{s}_k$, which follows from (1.69). Using this norm and weighting matrix, the unique solution of (1.76) is given by

$$H_{k+1} = (I - \frac{\mathbf{y}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}) H_k (I - \frac{\mathbf{s}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}) + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}. \tag{1.79}$$

This formula is called the DFP(Davidson, Fletcher and Powell) updating formula [21]. Since

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \epsilon_k \mathbf{p}_k = \mathbf{x}_k - \epsilon_k H_k^{-1} \nabla f_k = \mathbf{x}_k - \epsilon_k B_k \nabla f_k,$$

it is useful to derive the updating formula for the inverse Hessian approximation B_k

$$B_{k+1} = B_k - \frac{B_k \mathbf{y}_k \mathbf{y}_k^T B_k}{\mathbf{y}_k^T B_k \mathbf{y}_k} + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}, \tag{1.80}$$

which is found by making use of the Sherman-Morrison-Woodbury formula.

Instead of imposing an additional condition on H_{k+1}, one can impose the same closeness condition on its inverse B_{k+1}, which gives the following problem

$$\min_B \|B - B_k\| \tag{1.81a}$$

subject to

$$B = B^T,\ B\mathbf{y}_k = \mathbf{s}_k,\ \text{and } \mathbf{v}^T B\mathbf{v} > 0 \text{ for all } \mathbf{v} \neq 0, \mathbf{v} \in \mathbb{R}^n. \tag{1.81b}$$

Using again the weighted Frobenius norm with the average Hessian as the weight matrix satisfying $W\mathbf{s}_k = \mathbf{y}_k$, the unique solution is given by

$$B_{k+1} = (I - \frac{\mathbf{s}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}) B_k (I - \frac{\mathbf{y}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}) + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}, \tag{1.82}$$

which is called the BFGS updating formula. Again, using the Sherman-Morrison-Woodbury formula the inverse of B_{k+1} can be found

$$H_{k+1} = H_k - \frac{H_k \mathbf{s}_k \mathbf{s}_k^T H_k}{\mathbf{s}_k^T H_k \mathbf{s}_k} + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}. \tag{1.83}$$

1.5. Multi-scaling; Slow and fast modes

Molecular dynamics is an useful technique for studying (thermo-) dynamics of physical systems like proteins and other biomolecules. However, using conventional MD it is nearly impossible to have access to all the time scales of motion with atomic resolution within reasonable time. The simulation performance of molecular systems, with various length and time scales, is dictated by the smallest scale in the model description. The small time step size limits the study of important conformational changes, which are determined by the slow modes of the system. For improving the simulation performance the basic idea is to separate the dynamics into fast and slow modes and integrate the slow motions with a larger time step. In Figure 1.2 the contours of a system of 2 coordinates ($\Phi(x_1, x_2)$), labeled x_1 and x_2 , are projected in a 2D plot.

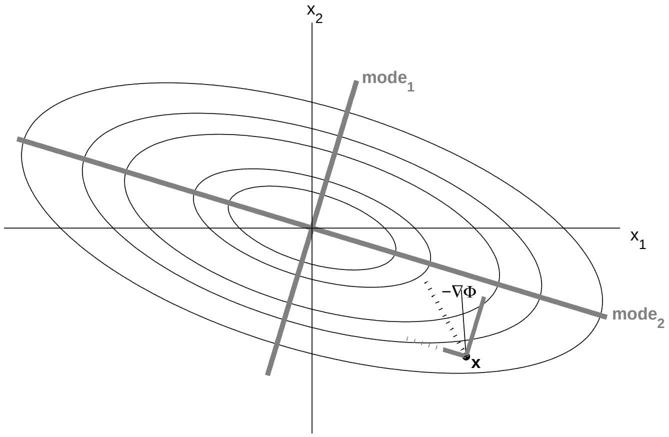


Figure 1.2: *Illustration of the different modes of a system.*

The minimum is located in the center and the normal mode directions are given by the gray lines, labeled mode_1 and mode_2 in Figure 1.2. Moving along a normal mode is related to the curvature of the potential along the mode direction. The normal modes and their frequencies are given by the eigenvectors and eigenvalues of the Hessian. The frequency is related to the curvature of the potential along the direction of the mode. Slow modes correspond to eigenvectors with small eigenvalues and fast modes correspond to eigenvectors with large eigenvalues. The normal mode directions are independent, which enables us to write any motion as a linear sum of the normal modes. In Figure 1.2 we have plotted the force $-\nabla\Phi$ at location $\mathbf{x}$. The force is a compound motion of the slow (mode_2) and fast (mode_1) modes. Although the model is simplistic, it shows that the amplitude along the slow mode is smaller than the amplitude along the fast mode. By artificially increasing the amplitude along the slow mode, a larger motion toward the minimum is made, i.e. a faster decrease of the energy potential. In Figure 1.2 we amplified (grey dashed line along mode_2) the slow mode of the force at $\mathbf{x}$, such that the amplitude for the slow mode becomes the same as the amplitude of the fast mode. The resulting total force, sum of the amplified slow mode and the original fast mode, is also given (black dashed line). Clearly amplifying the slow mode here results in a larger step toward the minimum.

As shown in the above example, partitioning the system into dynamics with fast modes and dynamics with slow modes allows for efficient propagation of the slow dynamics. Among popular approaches in separating modes of motion are *quasi-harmonic analysis* [22], *molecule optimal dynamic coordinates* [23] and *essential dynamics*[24].

In this thesis we propose a method which amplifies the slow modes of the system, which effectively means integration of the Langevin equation with a larger time step. The proposed method does not require any explicit partitioning of the space into dynamics with fast modes and dynamics with slow modes. Hence, the title of this thesis *Stochastic quasi Newton method for molecular simulations* could also be generalized into *Automated multi-scaling method with Langevin Dynamics for molecular simulations*.

CHAPTER 2

Improved configurational space sampling: Langevin dynamics with alternative mobility

We present a new and efficient method for determining optimal configurations of a large number (N) of interacting particles. We use a coarse-grained stochastic Langevin equation in the overdamped limit to describe the dynamics of this system, and replace the standard mobility by an effective space dependent inverse Hessian correlation matrix. Due to the analogy of the drift term in the Langevin equation and the update scheme in Newton's method, we expect accelerated dynamics or improved convergence in the convex part of the potential energy surface Φ. The stochastic noise term, however, is essential for proper thermodynamic sampling, but also allows the system to access transition states in the concave parts of Φ. We employ a Davidon-Fletcher-Powell (DFP) method for updating the local mobility matrix. Quantitative analysis for one and two dimensional systems shows that the new method is indeed more efficient than standard methods with constant effective friction. Due to the construction, our effective mobility adapts high values/low friction in configurations which are less optimal and low values/high friction in configurations that are more optimal.

2.1. Introduction

Condensed phases, whether liquid, glassy, or crystalline, owe their existence and measurable properties to the interactions between the constituent particles. These interactions are comprised in a potential energy function $\Phi(\mathbf{r}_1, ..., \mathbf{r}_N)$ that depends on the location $\mathbf{r}_i$ for each of those (N) particles. Material-specific contributions to this potential energy function constitute a multi-dimensional ($3N$ for structureless particles) potential energy landscape. Various static and dynamic phenomena in condensed phases emerge as manifestations of the complex topography of this hypersurface Φ [25].

An issue concerns the presence and determination of special points on the Φ hypersurface, in particular the minima and saddle points. Minima correspond to stable particle configurations, such that any small distortion will result in a restoring force to the undistorted arrangement. The global minimum is related to the state of the system at zero Kelvin, provided that the system is cooled slowly enough to maintain thermal equilibrium. In general, several minima with a substantial variation in depth are arranged in a complex pattern throughout the configuration space; for a single-component system rather general arguments show that the total number of minima scales as $N\exp(\alpha N)$ [26], where $\alpha > 0$ depends on the chemical nature of the system considered. Each minimum is enclosed in its own basin, consisting of all points $\mathbf{R} = (\mathbf{r}_1, ..., \mathbf{r}_N)$ in the direct vicinity of the minimum where $\Phi(\mathbf{R})$ is monotonic. Saddle points can be found on the boundary between contiguous basins, and represent the transition states of the system.

Standard numerical optimization techniques for the determination of these special points on the multi-dimensional hypersurface have several drawbacks: 1) in practice, nonlinear optimization methods (for instance Monte Carlo methods (MC)) are compute intensive due to rather poor convergence, 2) deterministic algorithms are not devised to sample multiple basins/minima and access them via transition states. Moreover, general numerical methods are not based on fundamental physical laws underlying the time evolution of the system and obscuring the relation between the simulation pathway and the dynamical phenomena that one would like to capture. In molecular dynamics (MD), the pathway is prescribed by the classical Newtonian equations of motion, which incorporate forces specified by Φ. Although MD is commended as an exact method and in general provides an adequate description of the particle dynamics, the length of the pathway in configuration space is seriously limited by the restriction of the time integration step to very small values, caused by the presence of high frequency modes. For modeling of phenomena on a long time and/or large length scales, as well as the configurational sampling of large dense systems, coarse-grained approaches are a reasonable alternative to MD. Coarse-grained descriptions reduce the degrees of freedom in the system by employing timescale separation, and take into account only the stochastic properties of the rapidly varying quantities. Stochastic dynamics techniques (of which Brownian dynamics is the most simple form [27]), use this approach to represent the presence of solvent by a stochastic and a frictional force in the Langevin equation for the solute. Moreover, one can argue that hydrodynamic modes become irrelevant for the dynamics of dense and strongly structured systems [28], and assume that the velocity distribution plays no relevant role. In particular for dense systems, the rapid flow of momentum and energy due to particle-particle collisions in the Newtonian dynamics is given by the bath in the Brownian dynamics. In the equilibrium limit, as a very long time average of the dynamic evolution, the Newtonian and the Brownian equations of motion should give the same result and be equal to those of the equilibrium statistical ensemble.

Our aim here is to model the Brownian behavior of N particles interacting via a pairwise potential. This potential could be a Coulomb interaction, a Lennard-Jones type interaction or simply a hard sphere repulsion. We employ microscopic equations of motion and include information about the local curvature of the potential energy hypersurface Φ in the Langevin equation describing the Brownian dynamics. The configuration-space sampling path resulting from our method is unconditional, and not necessary physically realistic. There are conceptual similarities to some efficient MC techniques [29, 30].

This chapter is structured as follows. First we introduce our method and discuss the details and numerical implementation. Following, we compare the performance of this method for a few lower dimensional problems to standard methods. We discuss the properties and results obtained by the new method in details. The focus here is on the efficiency, the sampling path, and the numerically constructed sampling ensemble. The application of this method to systems of higher dimensionality is dealt in the next chapters.

2.2. Our method with alternative mobility tensor

As mentioned in the introduction, we consider the Brownian behavior of N particles, interacting via a pairwise potential, in the high friction limit. The dynamics of this system is described by the general position Langevin equation, written in Ito form as [31]

$$\mathrm{d}\mathbf{x} = [-M(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T \nabla \cdot M(\mathbf{x})]\,\mathrm{d}t + \sqrt{2k_B T} L(\mathbf{x})\mathrm{d}W(t). \tag{2.1}$$

The second term on the right hand side of equation (2.1) is the compensation for the flux caused by the random force [31]. Here Φ is the potential energy, k_B is the Boltzmann constant, T the temperature and $M(\mathbf{x})$ is the space dependent mobility tensor with

$$M(\mathbf{x}) = L(\mathbf{x})L(\mathbf{x})^T, \tag{2.2}$$

and $W(t)$ is the multivariate Wiener process with $\langle \mathrm{d}W_i(t)\mathrm{d}W_j(t)\rangle = \delta_{ij}\mathrm{d}t$. In the simulations, we use random numbers with Gaussian distribution for the Wiener process. For simplicity, we set $k_B = 1$ and render equation (2.1) in the dimensionless form, i.e. all variables and functions (e.g. the energy potential $\Phi(\mathbf{x})$) in equation (2.1) will be considered dimensionless. Further, we will write κ instead of $2k_B T = 2T$. The amplitude of the noise term is determined in agreement with the fluctuation-dissipation theorem. As a result, the corresponding Fokker-Planck equation gives rise to proper sampling, according to the Boltzmann distribution, see appendix.

In the standard approach, the mobility tensor is chosen as the identity I, giving rise to a scalar prefactor describing the friction due to the implicit solvent. Here we include space dependency via the second order derivative and consider the mobility equal to the inverse Hessian H^{-1}. As a consequence, the deterministic part of equation (2.1) (for $T = 0$)

$$\mathrm{d}\mathbf{x} = -\,[H(\mathbf{x})]^{-1}\,\nabla\Phi(\mathbf{x})\mathrm{d}t, \tag{2.3}$$

is similar to the update scheme in Newton's method

$$\mathbf{x}_{k+1} - \mathbf{x}_k = \Delta\mathbf{x} = -\alpha H_k^{-1}\nabla\Phi(\mathbf{x}_k), \tag{2.4}$$

where dt is the (infinitesimal) time interval, and α is an appropriate positive step size, often obtained by a line search method. In the following we use t as continuous variable and k as iteration variable. Newton's method is standard in unconstrained nonlinear optimization, i.e. methods that aim at minimizing a certain objective function $\Phi(\mathbf{x})$, $\Phi(\mathbf{x}) : \mathbb{R}^N \to \mathbb{R}$. The methods make use of a *quadratic model* in which $\Phi(\mathbf{x})$ is approximated on the k-th iteration by a Taylor series about $\mathbf{x}_k$,

$$\Phi(\mathbf{x_k} + \delta_k) \approx q_k(\delta_k) = \Phi(\mathbf{x}_k) + \delta_k^T\nabla\Phi(\mathbf{x}_k) + \frac{1}{2}\delta_k^T H(\mathbf{x}_k)\delta_k\ . \tag{2.5}$$

The displacement δ_k on iteration k follows as the minimizer of $q_k(\delta)$. A unique minimizer of $q_k(\delta)$ exists if and only if $H(\mathbf{x}_k)$ is positive definite, and Newton's method is only well-defined in this case. Usually, $H(\mathbf{x}_k)$ is positive definite for small δ. It can then be proved that the sequence $\{\mathbf{x}_k\}$ converges, and that the order of convergence is of second order (quadratic convergence). When $\mathbf{x}_k$ is remote from the local solution $\mathbf{x}^*$ Newton's method may not converge, and may not be defined (when $H(\mathbf{x}_k)$ is not positive definite). Prototype algorithms such as line search and trust region methods can be employed to avoid this problem.

Especially for large N or when evaluation of $\Phi(\mathbf{x})$ is expensive, explicit calculation of the Hessian or inverse Hessian is too demanding, and Newton-like methods are a good alternative. These methods are based on approximating the exact Hessian $H(\mathbf{x}_k) = H_k$ in (2.5) by $G(\mathbf{x}_k) = G_k$ (giving rise to $B_k = G_k^{-1}$ approximate of H_k^{-1} in (2.4)). The reduction of $\Phi(\mathbf{x})$, the descent property, is guaranteed if G_k (and therefore B_k) is positive definite. The rate of convergence depends on the second derivative information installed in B_k, and ranges from linear to quadratic. The simplest positive definite choice is $B_k = G_k = I$, which does not contain any curvature information, and gives rise to the well-known steepest descent (SD) method with slow linear convergence. An alternative choice for indefinite H_k^{-1} is $B_k = H_k^{-1} + D_k$, where D_k is a diagonal matrix with proper elements. The efficient Quasi-Newton (QN) method builds up second-derivative information by estimating the curvature along a sequence of search directions [32]. Each curvature estimate is installed in an approximate inverse Hessian B_k by applying a rank-one or a rank-two update. The most successful updates are the Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula for the Hessian update and the Davidon-Fletcher-Powell (DFP) formula for the inverse Hessian update, which are members of the wider Broyden class of rank-two updates. The Quasi-Newton method with the BFGS (or DFP) update formula is also known as the BFGS (or DFP) method.

Here, we aim at developing a general and efficient scheme for equilibrating systems with many particles (large N), using the stochastic Langevin equation (2.1) in discretized form. In analogy with Newton-like methods, we consider a mobility matrix M_k that is a good approximation of the inverse Hessian H_k^{-1} to avoid direct calculation of this large $3N \times 3N$ matrix. In particular, we use the DFP method to determine this approximate matrix $M_k = B_k$ (see the appendix for details about the DFP update scheme). For $T = 0$, our update scheme is equal to that of the DFP method. The DFP method is known to yield a positive definite matrix B_k if the Wolfe conditions are satisfied [33], which requires that the next step is in the descending direction and that the displacement is sufficient large. Moreover, in that case it shows global convergence for convex functions $\Phi(\mathbf{x})$ [34]. In contrast to the standard practice, we deal with potentials $\Phi(\mathbf{x})$ that are convex only in a relatively small subset of configuration space. Our method should be able to hop between different minimal states, that are separated by energy barriers, and not be too sensitive to the starting configuration $\mathbf{x}_0$. A similar problem is present in reaction path sampling: in some regions, especially in the vicinity of saddle points, the Hessian can be negative definite, and DFP update becomes ill-conditioned [35]. Recently, a number of alternative methods that do not necessarily always satisfy the descent condition, the symmetric rank one formula (SR1), Powell-symmetric-Broyden (PSB) method and Bofill's formula, were developed and applied for reaction path sampling. Calculated pathways between reactant and product minima, via optimized transition states, were found to be accurate for a number of problems [36]. Nevertheless, we need to maintain the positive definiteness of the mobility matrix M_k at all times, since the Cholesky decomposition in equation (2.2) requires this property.

As a consequence, the DFP method needs to be adapted to handle the cases where the Wolfe conditions are violated. For computational reasons, we want to avoid any type of preconditioning of the matrix B_k which requires the explicit calculation of eigenvalues. A possible general remedy to avoid ill-conditioning is to switch back to the steepest descent method, i.e. restart the DFP update from the initial guess $B_k = I$. A disadvantage of this restart is that it would disregard all curvature information that has been build up along the pathway in configuration space. Another solution is updating the Cholesky factor L rather than $B = LL^T$ itself, which ensures a positive definite B [37]. However, no practical advantage can be expected for ill-conditioned problems [38], apart from an assurance that B_{k+1} remains positive definite in the presence of round-off errors.

The key difference between (2.1) and standard DFP methods is the stochastic term that is added at every step to ensure correct Boltzmann weights ($\sim \exp(-\Phi/k_B T)$), which can be seen as an information-based correction to the DFP update with varying amplitude and direction. Hence, our update is a combination of a coherent and stochastic contribution. The coherent contribution is always in the descending direction, by virtue of DFP. The stochastic contribution allows for sampling in the vicinity of the coherent update, the extent of which is determined by thermodynamic properties, and on average gives rise to the important ascending updates and sampling of transition states. The efficiency of the coherent part, the DFP method itself, depends on the appropriateness of the curvature estimate. We choose to conserve the quadratic information that is accumulated in the sequence of expanding subspaces in the convex part of Φ (where B_k is positive definite) by the DFP method. We use a small (constant) step size and do not update B_k if the Wolfe conditions are violated, i.e. if the approximate curvature $\mathbf{y}_k^T \mathbf{s}_k \leq 0$, we take $B_{k+1} = B_k$. One can show that this choice does not affect the theoretical equilibrium distribution of states, which is the Boltzmann distribution. Since the pathway between different minima should pass through regions with relatively small Boltzmann weights, we expect fewer sampling in the part of the potential energy surface where B is not updated, and hence an increased efficiency of the update scheme in this region. In terms of physical properties, reduced sampling can be associated with accelerated kinetics and lower friction. We will illustrate the peculiarities of this choice by considering the one and two dimensional problems. Since this scheme ensures the positive definiteness of $M_k = B_k$ at all times, the decomposition in equation (2.2) can be obtained by the standard Choleski decomposition.

The calculation of the divergence of the mobility tensor in the second term of the right hand side of equation (2.1) is compute expensive. To avoid direct calculation of the divergence term Hütter and Öttinger [39] proposed the following version of the stochastic differential equation (SDE)

$$\mathrm{d}\mathbf{x} = \left[-M(\mathbf{x})\nabla\Phi(\mathbf{x})\right]\mathrm{d}t + \frac{1}{2}\left[M(\mathbf{x}+\mathrm{d}\mathbf{x})M(\mathbf{x})^{-1} + I\right]\sqrt{2k_BT}L(\mathbf{x})\mathrm{d}W(t). \tag{2.6}$$

Equation (2.6) suggests the use of the predictor-corrector method in the numerical evaluation. The corresponding numerical scheme becomes

$$\begin{aligned}\Delta\mathbf{x} \quad = \quad & -\frac{1}{2}\left[M(\mathbf{x}+\Delta\mathbf{x}^p)\nabla\Phi(\mathbf{x}+\Delta\mathbf{x}^p) + M(\mathbf{x})\nabla\Phi(\mathbf{x})\right]\Delta t \\ & +\frac{1}{2}\left[M(\mathbf{x}+\Delta\mathbf{x}^p)M^{-1}(\mathbf{x}) + I\right]\sqrt{2k_BT}L(\mathbf{x})\Delta W_t,\end{aligned} \tag{2.7}$$

where ΔW_t is the Wiener increment with $\langle\Delta W_t\rangle = 0$ and $\langle\Delta W_{t_i}\Delta W_{t_j}\rangle = \delta_{ij}I\Delta t$ and $\Delta\mathbf{x}^p$ is the predictor step

$$\Delta\mathbf{x}^p = -M(\mathbf{x})\nabla\Phi(\mathbf{x})\Delta t + \sqrt{2k_BT}L(\mathbf{x})\Delta W_t. \tag{2.8}$$

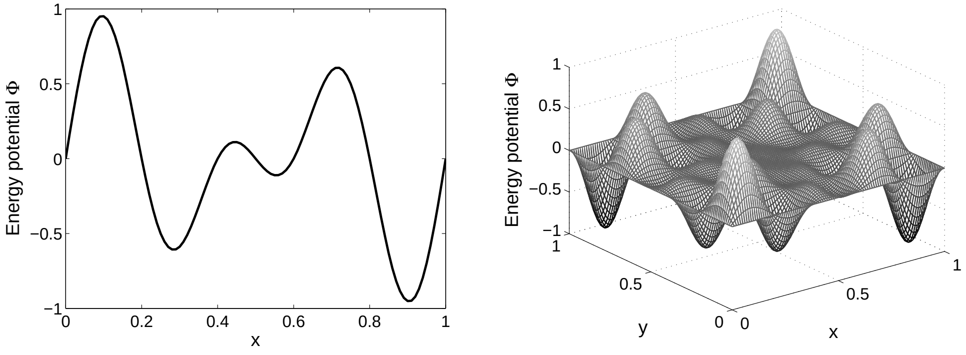
The updated mobility tensor obtained by the DFP method, $M(\mathbf{x}_k) = B(\mathbf{x}_k)$, guarantees the existence of the inverse in equation (2.7). The integration scheme is weakly convergent to first order in the time step Δt. This method replaces the calculation of the divergence term and is clearly favored to the direct calculation (except in cases where $\nabla \cdot M(\mathbf{x})$ is given in closed form). Details about the DFP update, space and time dependent, can be found in the appendix.

2.3. Results and discussion

In order to compare our method to standard approaches for which the performance is known analytically, we restrict the application of the new method to one and two dimensional systems. In particular, these energy potentials are not related to any real physical system.

2.3.1. Simulated sampling distributions

First, we verify the thermodynamic accuracy, i.e. whether our method indeed samples according to the Boltzmann distribution $\mathcal{N}\exp\left(-\Phi/k_BT\right)$. The considered energy landscapes, containing multiple minima, are shown in Figures 2.1a and 2.1b, and are periodic in space (we use periodic boundary conditions) in order to obtain a more differentiated sampling. The simulated sampling distributions were calculated from the simulation results by a binning routine, with small bin width h_b or area h_b^2. To obtain Figures 2.2 and 2.3b, the number of samples obtained in each bin were divided by the total number of samples $K = 10^6$. In Figure 2.2, the simulated distribution (for $\Delta t = 0.01$ and $\kappa = 1$) is compared to the expected distribution function. The expected distribution is the stationary solution of the Fokker Planck equation. There is no driving force in equilibrium, hence there is zero flux. This is equal to the Boltzmann distribution for our choice of the drift and noise term [40]. It can be observed that the simulated distribution matches the theoretical distribution reasonably well,



(a) *1-D energy landscape* **(b)** *2-D energy landscape*

Figure 2.1: *The considered low dimensional sample cases.*

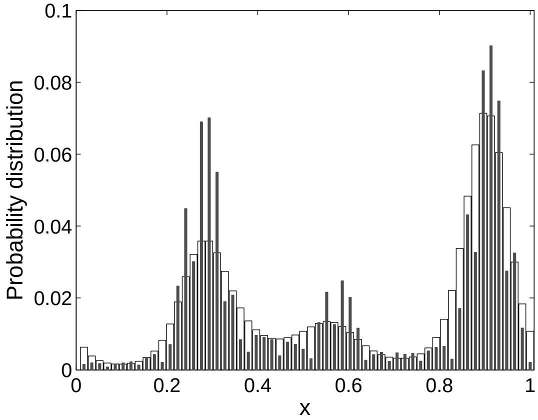
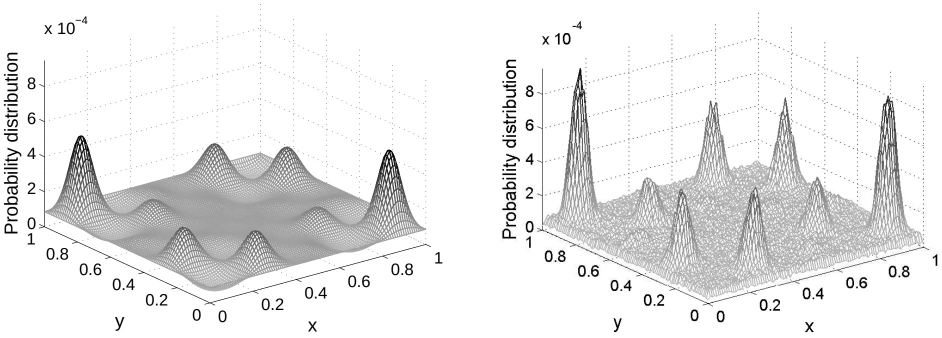


Figure 2.2: *Comparison of numerically calculated (full bars) and analytical (open bars) probability distributions. The numerical results are obtained using our sampling method with* $\kappa = 1$ *and* $\Delta t = 0.01$.

qualitatively but also quantitatively. The oversampling of the maxima, and the global maximum of the distribution in particular, is due to finite sampling and the peculiarities in the update scheme for B_k. On average, we also observe undersampling of states with lower Boltzmann weights. At a later stage, we discuss how our choice for the mobility $M_k = B_k$ contributes to these effects. For the two dimensional case, the theoretical distribution (Figure 2.3a) and the simulated distribution (Figure 2.3b) match rather well, but show even somewhat more pronounced over- and undersampling than in 1D. Overall, we conclude that the sampling is accurate in both cases. For larger problems, i.e. large N, the efficiency of our new method is of importance. Here, we compare the mean first passage times (MFPTs) obtained by our mobility to that of (2.1) for M_k is constant. The latter choice gives rise to the standard Langevin equation for particles motion in the overdamped situation, with a constant friction coefficient. Instead of the one dimensional potential in Figure 2.1a with multiple minima, we use a double-well potential. The advantage of this double-well potential is that an analytical expression for the MFPT is known [41]. We note that the one dimensional case is special, as any sampling path between different (quasi) equilibrium states automatically requires crossing the potential energy barrier. For higher dimensional energy surfaces, high energy barriers can be avoided in favor of lower energy transition states or saddle points, and the performance of the method crucially



(a) *Analytical probability distribution* **(b)** *Calculated probability distribution*

Figure 2.3: *Comparison between analytical distribution function and the calculated distribution function with* $\kappa = 1$ *and* $\Delta t = 0.01$.

depends on the ability to find the optimal route.

2.3.2. Mean first passage times

We consider the mean escape time from a 1-D well, see Figure 2.4. By this we mean the first passage time from a to x, where $x \to b$. The mean first passage time (MFPT) $\langle t \rangle$, defined as the average time needed from a well to the top of the hill is a quantitative measure for the average transition time between the two different (quasi) equilibrium states. Numerically, we performed a large number of simulations (1000), all starting in point a in Figure 2.4 using different noise seeds. The simulations were finalized upon passing b for the first time, i.e. for $k = \tilde{k}$ such that $x_k > b$. The simulation times $\tilde{k} \cdot \Delta k$ (in this article, k and t can be interchanged, and both represent time) were later averaged to obtain the simulated $\langle t \rangle$. As mentioned, the simulated MFPT can be directly compared to the MFPT for the standard Langevin equation, equation (2.1) with $M(\mathbf{x}) = M$ is constant. We make our comparison scale invariant by choosing the constant mobility $M = H^{-1}(x_{min})$, where x_{min} is the location of the starting point. The corresponding dimensionless position Langevin equation is

$$\dot{x} = -M\nabla\Phi(x) + \zeta(t), \tag{2.9}$$

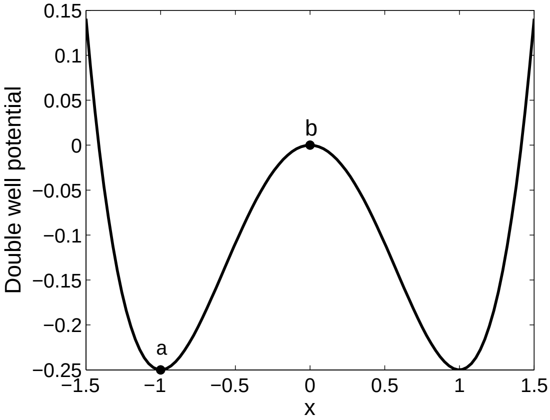


Figure 2.4: *The double-well potential considered;* $\Phi(x) = -\frac{1}{2}x^2 + \frac{1}{4}x^4$.

and the analytical representation for the MFPT is given by the integral formula

$$\langle t\rangle = \frac{2}{\xi}\int_a^b \mathrm{d}y \exp\left(\frac{2}{\kappa}\Phi(y)\right)\int_{-\infty}^{y} \mathrm{d}z \exp\left(-\frac{2}{\kappa}\Phi(z)\right), \tag{2.10}$$

where ξ is the noise intensity of ζ. Again, a is the initial condition and b is the final state [42]. For the double-well potential shown in Figure 2.4, defined by $\Phi(x) = -\frac{1}{2}x^2+\frac{1}{4}x^4$, we can numerically integrate equation (2.10) and derive theoretical values of the MFPT for varying κ. Alternatively, one could also directly use the well-known Kramers equation [42] to calculate theoretical MFPTs, but this explicit equation is only strictly valid in the limit of high friction. The theoretical values (2.10) and the simulated MFPTs, with $M(x_k) = B(x_k)$, are compared in Figure 2.5 for varying κ. Additionally we have tested our implementation by evaluating our scheme for constant mobility $M = M(x_{min})$. We observe that the simulated MFPTs for this mobility are in very good agreement with the numerically integrated theoretical predictions of (2.10). An important observation is that the simulated MFPTs for $M(x_k) = B(x_k)$ are significantly smaller than the MFPTs for the constant mobility ($M = M(x_{min})$) for the whole κ range. The efficiency increases significantly for small κ (or large κ^{-1}, see Figure 2.5). Since the contribution of the stochastic term is proportional to κ, it is a clear indication that the incorporation of our new mobility, $M_k = B_k \approx H_k^{-1}$, in the deterministic part of (2.1) allows for a much more efficient crossing of energy barriers. This finding is important, since one could think on beforehand that the effect

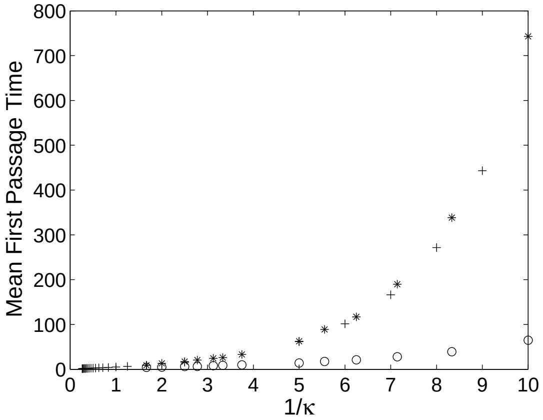


Figure 2.5: *Mean first passages times (MFPTs) of the double-well potential, calculated for different noise amplitudes. The plusses (+) are the MFPTs calculated by numerically integrating the analytical expression (2.10), with constant mobility* $M = H(x_{min})^{-1}$*. The MFPTs found by our scheme (2.1) using a constant* $H(x_{min})^{-1}$ *(∗) and space dependent* B_k *(○) are shown as well. The step size* $\Delta t = 0.01$.

of different mobility is rather small in one dimension, where M is a scalar variable. In particular, using $M = H^{-1}(x_{min})$ or $M = B(x_k)$ in equation (2.1) gives rise to the same update for the deterministic part, apart from a space dependent pre-factor. For higher dimensional energy surfaces, the search direction itself will play an important role. We have earlier noticed that, apart from the pre-factor, the deterministic part of (2.1) for $M = M(x_{min})$ is analogous to the update in the standard steepest descent method, independent of the dimensionality of the system. From transition state theory for chemical reactions it is well-known [43] that the QN method is much better suited for locating saddle points or transition states than the steepest descent method. Hence, we expect that especially for high-dimensional systems the choice $M_k = B_k$, related to the QN method, will lead to an improved sampling compared to $M_k = M(x_{min})$, that is related to the steepest descent method. In conclusion, we expect a further reduction of the transition times (or MFPTs) for the new method and large N.

We have omitted error bars in Figure 2.5 and calculated the standard deviation of the MFPTs of the simulations. We found that the ratio of standard deviation to the mean is around one for each of the mobility tensors. This indicates that the MFPTs are exponentially distributed.

2.3.3. The update scheme for mobility

Finally, we turn to our update scheme for $M(\mathbf{x}_k) = B_k$ and consider how our procedure to maintain positive definiteness affects the sampling and efficiency. In particular, we only use the DFP update for B_{k+1} if the Wolfe conditions are satisfied, otherwise $B_{k+1} = B_k$. The particular potential surface, the one dimensional double-well potential $\Phi(x)$, $H^{-1}(x)$ (only for positive values) and the average B as a function of Cartesian co-ordinate x are combined in Figure 2.6. Since we again consider the one dimensional potential, all functions have scalar values, and we use the values from the many simulations that were carried out for the calculation of the MFPT. Here and further on, all averages are calculated by a binning routine. For each bin, the average is calculated over all $B(x)$ within the bin.

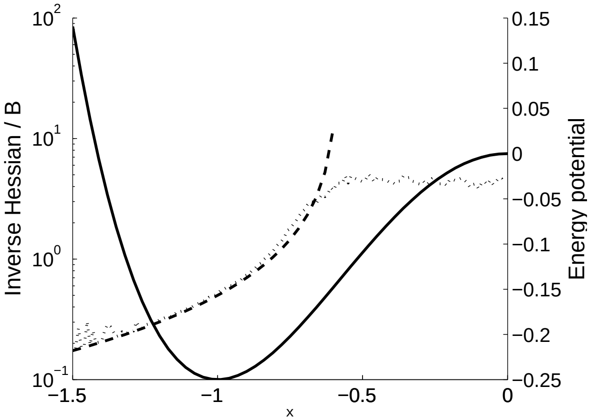


Figure 2.6: *The double-well potential (solid line), the analytical inverse Hessian (dashed line) and calculated average mobility (dotted line) for* $\kappa = 0.2$ *and* $\Delta t = 0.01$.

In the convex region, where H (and thus H^{-1}) is positive, our update scheme clearly yields a good approximate for H^{-1}, as expected by virtue of the DFP method. Only close to the initial value a the average B is less accurate. This is due to the initial guess $B_0 = I$ and build-up of curvature information. In the concave region, $H^{-1}(x)$ is negative, and hence there is no update in this region, i.e. $M(x_{k+1}) = B_{k+1} = B_k$. The mobility remain unchanged as long as x lies in the concave region. Only when the update crosses over to a convex region again, the mobility is updated. We refer to the boundary limit the region, where the DFP update is accepted or rejected as the cross-over point x^c. By virtue of our method the location of this cross-over point depends on the noise amplitude and the temperature T. We determine the average $B(x^c) = B^c$ value both numerically and by a simple theoretical estimate. Consider $x_k = x^{pre}$ in the convex region and $x_{k+1} = x^{post}$ in the concave region, and x^∞ the inflection point, i.e. the location where $H = 0$. To determine x^{pre} we consider the following equation

$$\begin{aligned}(\Delta x)^2 &= (x^{pre} - x^{post})^2 \\ &= 4(x^{pre} - x^\infty)^2, \end{aligned} \tag{2.11}$$

where the latter equation is obtained by using $x^{pre} = x^\infty - \frac{1}{2}\Delta x$. Equation (2.11) can

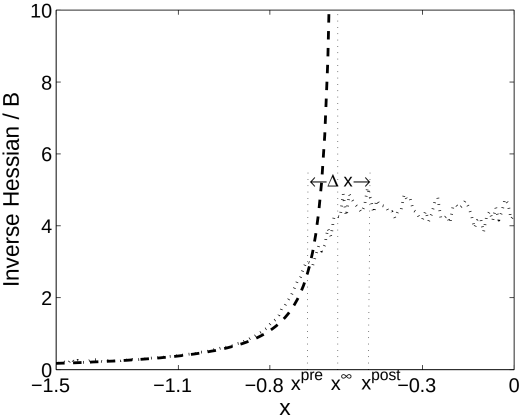


Figure 2.7: *Analytical inverse Hessian (dashed line) and the calculated average mobility (dotted line) with* $\kappa = 0.2$ *and* $\Delta t = 0.01$. *In addition we include a schematic view of* x^{pre}, x^{post} *and the asymptote at* $x = x^\infty$, *used to estimate the mobility when the Hessian* H *is negative. The actual position of* x^{pre} *and* x^{post} *depends on the noise amplitude.*

be rewritten as

$$4(x^{pre} - x^\infty)^2 = \kappa H^{-1}(x^{pre})\Delta t, \tag{2.12}$$

which becomes a simple fourth power equation, since $H(x)$ is a quadratic function in this special case. Because $H(x)$ and x^∞ are known, equation (2.12) can easily be solved analytically, which gives the theoretical value of x^{pre}. We compare the theoretical estimate $H^{-1}(x^{pre})$ and the numerically obtained average B^c in Figure 2.8.

Since we consider the average B^c, we loose information about the individual simulation pathways. In particular, some pathways cross the inflection point multiple times before reaching the top b, and as a result the location of the cross-over point, and hence the value of B^c, varies in time. We try to include these details in the averaging procedure by distinguishing between *all* pathways and pathways that cross the inflection point only *once* (to be called first-time passing in the remainder). The dotted line in Figure 2.8 shows the average B^c after first time passing only, and the dashed line shows the average B^c using all simulated pathways. As expected, the value of B^c after first time crossing is smaller than the value of B^c using all pathways, and for all graphs the value of B^c increases with decreasing noise amplitude κ. This behavior can be explained. First of all, one should realize that the *average* distance h between the cross-over and the inflection point is determined by the average stochastic contribution to the search direction, and therefore by the noise-amplitude in the stochastic term in (2.1). The distance h will decrease for decreasing κ (or increasing κ^{-1}), and since H^{-1} is a monotonically increasing function in this interval, the constant value of B^c will increase (see the relevant figures). Pathways that cross back from the concave to the convex region are more likely to also sample the interval between the first-time cross-over point and inflection point, and subsequently the positive definite B^c is higher on average, following the same arguments. We observe that all graphs in Figures 2.8 follow the same trend and are therefore equal, except for a scaling factor. We see that our choice for the mobility-update in the concave region results in a constant mobility B^c, with an average value that increases with decreasing κ. This is a convenient property in general, since high mobility or equivalent low friction gives rise to an acceleration of the scheme, and smaller residence times in this part of the energy potential landscape. Although the potential energy surface is different, this finding explains the oversampling of the maxima and undersampling of lower regions after a finite number of simulations steps, when compared to the theoretical Boltzmann distribution $\mathcal{N}\exp(-\Phi/k_B T)$ in Figure 2.2 (for $\kappa = 1$). For a decreased temperature or noise amplitude, the regions associated with low Boltzmann weights become even more pronounced, and standard Monte Carlo (MC) methods experience a critical slowing down and very long transition times. Multicanonical ensemble methods have been developed to enhance the behavior in these regions, by manipulating the sampling distribution [44]. In our method, however, the distribution itself is left unchanged, and does not have to be calculated explicitly. With decreasing temperature, the approach of the inflection points is slowed down (see Figure 2.5), but crossing this point is followed by ’tunneling’ through these regions due to particular features

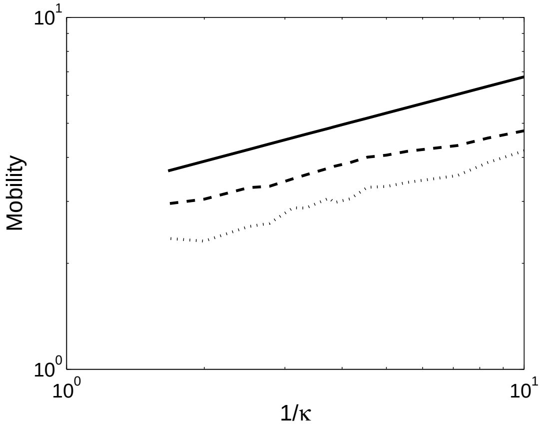


Figure 2.8: *Comparison of the analytically estimated mobility $H^{-1}(x^{pre})$ (solid line) with from simulation obtained B^c for the mobility in the concave region of the double-well; the average B^c after passing the inflection point (dashed line) and the average B^c after the first time passing the inflection point (dotted line), with $\Delta t = 0.01$.*

of the update scheme.

To be complete, we have also considered the properties of our scheme (2.1) for the potential energy surface with multiple extrema (see Figure 2.1a). In Figure 2.9, we compare the analytic $H^{-1}(x)$ for this potential energy surface and the average B for two different temperatures: $\kappa = 0.2$ and 1. We observe that the average B associated with the lower temperature $\kappa = 0.2$ is a very good approximate of the (analytic) inverse Hessian in the convex region, and almost constant outside this region. This finding is very well in accordance with the double-well case. We also observe that the large energy barriers in the potential energy surface cannot be overcome for this relatively small κ. For higher κ, the scheme samples the whole interval $x \in [0, 1]$, which can also be observed from the simulated distribution in Figure 2.2. The average mobility B, however, appears to be noisy. At first this may seem surprising, as the DFP method is known to approximate the actual $H^{-1}(x)$ in one step for a 1D function $\Phi(x)$, if $\Phi(x)$ is a (nearly) quadratic function. Moreover, the (almost) constant value B^c in the concave part of the landscape is higher for $\kappa = 1$ than for $\kappa = 0.2$, which seems in contradiction with the conclusion drawn for the double-well potential. Looking at $B = B(x)$ in somewhat more detail, we may consider B a superposition of low and high frequency contributions. The shape of this low frequency contribution is very similar to that of the inverse Hessian in the convex region. In the concave region, it is a constant. Hence, we conclude that this contribution can be associated with the part of the pathway where the matrix B is frequently updated, and the DFP scheme provides a good approximate B of the actual $H^{-1}(x)$. The high frequency contribution gives rise to a shift of the average B towards higher values, when compared to $H^{-1}(x)$. A similar shift in the direct vicinity of x^{pre} can be observed for the double-well potential in Figure 2.7. Although these simulations were stopped when the pathway reached point b and here multiple extrema are sampled instead, the origin of this overestimation is the same. Due to the random displacements, the update can hop between convex and concave regions in the vicinity of the cross-over point. Since B in the concave part is taken as the value of the *last* positive definite update B^{last}, an update that hops back to the convex region at step k uses B^{last} to update B_{k+1}. Consequently, the calculation of the average $B(x)$ in the vicinity of the cross-over point includes a number of values B^{last} that may be inaccurate approximates of the inverse Hessian $H^{-1}(x)$. Moreover, the values associated with these last updates are likely to be higher (on average) than the actual inverse Hessians $H^{-1}(x)$ (see also the previous discussion for the double-well potential using only first-time passages and all pathways). Since the range of the random displacements scales with the noise amplitude, the affected range of $B(x)$ is also larger for larger κ. This explains the superposition of two contributions with a different character. This overestimation is observed in the whole range of $\Phi(x)$ and is due to the fact that the wells in Figure 2.1a are not well separated. We conclude that only a part of the sampling pathway is affected, and that the mobility in this part is artificially high. However, from the comparison in Figure 2.2 we see that the Boltzmann distribution is well reproduced by the numerical sampling, so the effect is rather minimal.

For 2D potential energy surfaces, the simulated distribution in Figure 2.3a has features that are similar to the ones observed in 1D (see the section about the simulated distribution of states), and we conclude that our choice of M_k is apparently also efficient for sampling transition pathways for $N = 2$. We note that this is an important finding, as for $N > 1$ pathways that cross energy barriers (maxima of the function Φ) can be avoided in favour of saddle points. The situation for higher dimensional systems ($N > 2$) will be considered in the next chapters.

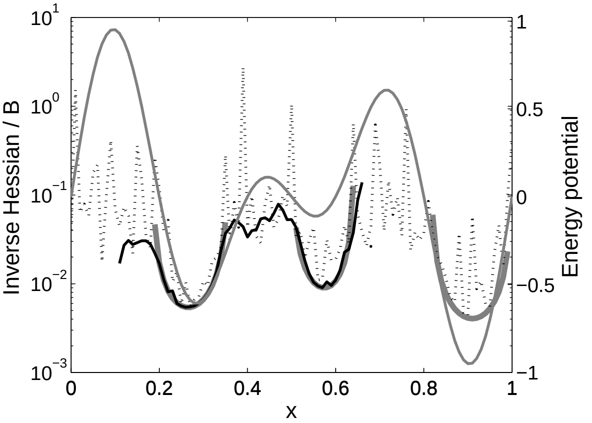


Figure 2.9: *The multi-state energy potential (solid gray line), the analytical inverse Hessian (thick solid gray line), with the average mobility B_k used in simulations. The dots represent B_k with $\kappa = 1$ (large noise) and solid black line B_k with $\kappa = 0.2$ (lower noise).*

2.4. Conclusion

We already discussed in detail the similarities between our displacement and the QN-displacement. The drift term in (2.1) contains the approximate correlation matrix B multiplied with the minus gradient of the energy potential $\nabla\Phi$ and dt. This expression is exactly the same as the displacement in the QN method. By only taking into account $-B\nabla\Phi \mathrm{d}t$ and the DFP approach for B guarantees the descending property of the method. Mathematically, the drift term guarantees quadratic convergence to the local minimum. Physically, one immediately notices that $-\nabla\Phi$ is the force, since Φ is an energy potential. If B is the identity matrix I, the displacement contribution of the drift term is exactly biased in the direction of the force (corresponds to the mathematical steepest descent method). For a general diagonal matrix, the biased direction is the force which magnitude is determined by the values of the diagonal. Using the approximate of the covariance matrix, the direction is determined by the correlation matrix B.

The stochastic term $\sqrt{2k_BT}L(\mathbf{x})\mathrm{d}W_t$ contains the temperature and the decomposition of the covariance matrix. The noise term has been obtained mathematically by satisfying the fluctuation-dissipation theorem. The temperature in the noise term determines the shape of the distribution. If the temperature is low, the distribution function will have sharply peaked maxima. The MFPTs have already shown that it can take a very long time to cross an energy barrier if the temperature is low. In the limiting case energy barriers will never be crossed, for $T = 0$ the method becomes the standard QN-method and the method will be trapped in the nearest (local) minimum. If the temperature is high, the noise term will dominate. The distribution function will be flattened, which means that each state becomes almost equally likely to be visited. Due to the high temperature the system is able to cross energy barriers and hop from one minimum to another. Too high temperature makes it difficult to identify the minima since there is no explicitly a favored state.

From the existing analytical expression for the MFPTs we already know that higher noise intensities, equivalent to the temperature in our case, will cause shorter MFPTs. We performed simulations to compare the numerical evaluation of the analytical expression for the MFPTs. Choosing the mobility as a constant gives us the same results as the analytical expression. Using our correlation matrix as mobility, the MFPTs are certainly shorter compared to the MFPTs obtained from a constant mobility $M = H^{-1}(x_{min})$ and thus implies a better performance of our method, based on the the cycles needed to find the MFPTs. To be more accurate, we need to consider the arithmetical operations needed in each cycle. It is obvious that our method needs more arithmetical operations in each cycle because of the update for the approximate inverse Hessian B_k ($o(n^2)$) and the Cholesky decomposition($o(n^3)$). In our one dimensional case, the MFPTs for $M = B_k$ is roughly one order lower than the MFPTs for $M = H^{-1}(x_{min})$ for small κ. Obviously the arithmetical operations in our sample case is limited since $n = 1$. For higher dimensions $n \gg 1$, the arithmetical operations may start to dominate if the performance of our method over the standard method does not improve as well. Existing knowledge of the efficiency of the QN and SD method indicates that this may actually be the case. However, future will concentrate on the use of limited memory methods for the iterative updates.

In short, our method combines the quadratic convergence of the drift part with the properties of statistical thermodynamics in the noise part. The update of the mobility tensor in the one dimensional double-well case shows that the mobility is large in concave regions of the energy potential. This corresponds with faster crossing over energy barriers, i.e. less friction. In the convex regions, the mobility is lower, corresponding to high friction. Hence, due to the constructed mobility the method over samples regions which are more likely and under samples regions which are less likely.

The temperature in our method has been kept constant during each of the simulations. One can imagine that the method can be improved by using the temperature as the tuning parameter of the system. For instance the temperature may be increased if the sampling path is pinned to a certain minimum for a long time. An increased temperature can help the method to cross over a certain energy barrier. Adaptive temperatures are a standard procedure in 'simulated annealing' [45].

Additionally, improvement of the numerical performance may come along by changing the time step Δt, that was considered constant throughout this study. This directly corresponds to changing the step size in a QN-method. A very common method is using linesearch to determine the step size. Other improvements, with respect to the DFP update scheme used here, are considered for larger dimensional systems and are studied in the remaining of this thesis.

CHAPTER 3

Stochastic Quasi-Newton molecular simulations

We report a new and efficient factorized algorithm for the determination of the adaptive compound mobility matrix B in a stochastic quasi-Newton method (S-QN) that does not require additional potential evaluations. For one dimensional and two dimensional test systems, we previously showed that S-QN gives rise to efficient configurational space sampling with good thermodynamic consistency (Chapter2). Potential applications of S-QN are quite ambitious, and include structure optimization, analysis of correlations and automated extraction of cooperative modes. However, the potential can only be fully exploited if the computational and memory requirements of the original algorithm are significantly reduced. In this chapter, we consider a factorized mobility matrix $B = JJ^T$ and focus on the nontrivial fundamentals of an efficient algorithm for updating the noise multiplier J. The new algorithm requires $\mathcal{O}(n^2)$ multiplications per time step instead of the $\mathcal{O}(n^3)$ multiplications in the original scheme due to Cholesky decomposition. In a recursive form, the update scheme circumvents matrix storage and enables limited-memory implementation, in the spirit of the well-known limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method, allowing for a further reduction of the computational effort to $\mathcal{O}(n)$. We analyze in detail the performance of the factorized (FSU) and limited-memory (L-FSU) algorithms in terms of convergence and (multiscale) sampling, for an elementary but relevant system that involves multiple time and length scales. Finally, we use this analysis to formulate conditions for the simulation of the complex high-dimensional potential energy landscapes of interest.

3.1. Introduction

The development of general-purpose methods for large-scale molecular simulation is an important scientific goal. The area of application is large and diverse, but one may think of typical phase separation phenomena in hard and soft matter systems, including particle cluster optimization, membrane formation, protein folding, micelles and polymer dynamics. The magnitude of the challenges involved cannot be underestimated, since in many chemical systems of relevance, the starting point for the model is necessarily atomic or molecular, while the emerging collective behavior on long length and time scales - determining the relevant material properties and function - essentially is not. A prime example is protein-folding, where the characteristic (length and time) scales associated with the smallest constitutive elements (electrons or atoms) deviate many orders of magnitude from those associated with the co-operative motion of protein domains, such as beta sheets or alpha helixes. Since the simulated system evolution, or, alternatively, the sampling-rate on the complex energy hypersurface, is dictated by the *smallest* scale in the model description, this co-operative motion remains inaccessible even on present-day supercomputers.

A common strategy to overcome some of these problems is by going from high to lower resolution, i.e., by averaging over the smallest degrees of freedom. Our starting point [46] is such a coarse-grained model, the general position Langevin equation describing the Brownian behavior of N interacting particles in the high friction limit, written in Ito form as

$$\mathrm{d}\mathbf{x} = -B\nabla\Phi(\mathbf{x})\mathrm{d}t + \sqrt{2kTB}\mathrm{d}W(t),, \tag{3.1}$$

for a molecular potential energy Φ depending on the system state $\mathbf{x} \in \mathbb{R}^{3N}$, with Boltzmann constant k_B, temperature T and $W(t)$ a multivariate Wiener process with $\langle \mathrm{d}W_i(t)\mathrm{d}W_j(t)\rangle = \delta_{ij}\mathrm{d}t$. The standard mobility B is constant and inversely proportional to the viscosity of the surrounding medium. Typically, such a (Brownian) dynamics model is simulated by an Euler scheme, over many, many time steps. The time step is determined by the fastest modes in the coarse-grained representation, associated with the steepest gradients in the energy landscape, and hence the scheme is again rather inefficient for slow modes associated with shallow gradients. Near phase boundaries these systems suffer from critical slowing down (we borrow the following arguments and notation from Dünweg [47]) due to the appearance of a very long correlation times. In these conditions, the system exhibits large correlated objects or 'critical clusters' of typical size ξ (the correlation length), which can be made arbitrarily large by means of some control parameter. As a general feature very many configurations are easily accessible, since the typical energy to change, create or delete such an object is, at most, of order of the thermal excitation energy $k_B T$. The key challenge in simulating these systems is that the physical dynamics is usually *local*, whereas the collective behavior is not. The rearrangement on a larger scale depends on the spread of information through the smaller scales (for diffusive dynamics, with rearrangement time $\tau \propto \xi^2$) and thus consumes increasingly more time for increasing object size ξ. In other words: the hypersurface associated with conformational (re)arrangement on a larger scale is relatively flat, and the dynamics dictated by (3.1) slows down extremely due to it local nature, the constant mobility and (almost) vanishing gradients. The system becomes increasingly soft and sluggish - the true hallmark of soft matter.

Common strategies to circumvent the limitations in accessible time and length scales are based on selecting and updating large length scales with artificially high rates. In molecular modeling of (bio)molecule dynamics, one often relies on enhanced sampling via parallel tempering/replica exchange [48, 49], simulated tempering [50], solute tempering [51], multicanonical molecular dynamics (MD) [52] and Wang-Landau [53]. Others, like metadynamics [54] and hyper-MD [55], introduce a bias on a small set of collective variables and recast the problem in terms of transition-state theory. Our S-QN method can also be seen as a method for enhanced sampling and (global) optimization [46]. Our focus here is its general applicability and on the multi-scale features, in particular the multiplicity of time steps that is *automatically* introduced by including curvature-information of the *unbiased* potential energy hypersurface. Our approach is essentially a real-space generalization of existing accelerated algorithms that use filtering for the separation of different length/time scales [56–58]. This Fourier acceleration (FA) technique [47] attempts to renormalize the characteristic times associated with different (Fourier) modes by using a mass matrix as a preconditioner to the forces in the Fourier domain. As such, it enables a *multiplicity* of time steps. Consequently, the determination of an appropriate mass matrix is key to the success of this technique. The mass matrix should be positive definite (due to the appearance of a square root in the noise term), and is often regularized to avoid problems associated with very small wave vectors. The renormalizing mass matrix can be determined analytically for a purely Gaussian model (a quadratic potential), where Fourier modes completely decouple and the integration can be carried out independently. For this model Hamiltonian, FA has indeed been shown to completely eliminate critical slowing down [59]. For general Hamiltonians with higher order terms, different modes may be *coupled*, and preconditioning with this mass matrix can easily fail. In particular, it is *a priori* uncertain if FA will work at all [60] and FA is also known to suffer from discretization artifacts [61]. We explicitly note that following or (re)constructing the actual "physical" dynamics of the system is not our purpose. In this sense, the S-QN method is very similar to FA and many of the other methods on different levels of description that are aimed at accelerated or enhanced sampling of energy landscapes. By construction, however, the characteristics of large-scale dynamics and important correlations will always be directly accessible.

We first clarify the central idea of S-QN. For simplicity, we omit the spurious drift term [46]. We consider one of the simplest systems possible, a Harmonic potential $\Phi(x) = \frac{h}{2}x^2$ ($x \in \mathbb{R}$), with h (in J/m^2) the force/spring constant. After introducing a second differential equation for the adaptive mobility B, the Langevin equation for this system is given by [46]

$$\mathrm{d}x = -Bhx\mathrm{d}t + \sqrt{2k_BTB}\mathrm{d}W(t) \qquad \text{initial state} \tag{3.2}$$

$$\frac{\mathrm{d}B}{\mathrm{d}t} = -B + \frac{1}{h} \tag{3.3}$$

$$B(0) = 1 \tag{3.4}$$

We note that the second equation is only used for the purpose of illustration: in the S-QN method, the constant inverse Hessian of Φ, $1/h$, is recursively determined using QN methodology. From (3.2), it is clear that the initial behavior at $t = 0$ is the *same* as for (3.1), i.e., Langevin dynamics with a constant $B = 1$. In this stage, the noise is decoupled from the energy landscape, and only the drift term acknowledges the local gradients on the potential energy hypersurface. In the stationary state ($B = 1/h$), however, (3.2) simplifies to

$$\mathrm{d}x = -x\mathrm{d}t + \sqrt{2\frac{k_BT}{h}}\mathrm{d}W(t) \qquad \text{stationary state} \tag{3.5}$$

We observe reversed roles: the drift term is no longer dependent on the gradient, but the random displacement is strongly dependent on the gradient through the proportionality to $\sqrt{1/h}$, and will therefore decrease in magnitude for increasing h. In addition, depending on the value of h, the drift term in (3.5) gives rise to an acceleration ($h < 1$) or slowing down ($h > 1$) compared to (3.1), or, alternatively, an effective scaling of the time by $1/h$. More general, the method was designed to automatically apply dense sampling in narrow basins with steep gradients containing minima and larger sampling steps in almost flat parts of the energy hypersurface where the gradients almost vanish. The noise term facilitates the escape from basins [46]. This differentiated sampling rate (for fixed dt) is obtained by acknowledging the topography of the hypersurface via curvature information. Hence, efficient incorporation of proper curvature information is vital, and we previously showed [46] that the standard QN framework for numerical optimization provides such methodology. In particular, for a constant time step dt, the inverse Hessian is iteratively constructed by a Broyden-Fletcher-Goldfarb-Shanno (BFGS) method using only gradient information in subsequent sampling points. However, we recognize that for the target systems involving multiple scales or, equivalent, large n, memory requirements and/or the computational load can still become limiting for the application of the S-QN method. The Cholesky factorization, necessary for computing the noise term [46], represents a considerable ($\sim \mathcal{O}(n^3)$) computational burden for each iteration. Storage may become an additional burden, since several $n \times n$ matrices should be updated and/or stored at each step. Consequently, algorithmic improvements that leave the general properties of the method unaffected but substantially *reduce* the storage and computational requirements are of great importance for the value of S-QN as an efficient general-purpose simulation method. Here, we focus on the derivation of such new and efficient algorithms. Factorized QN methods using triangular matrices have been considered, but these methods were primarily designed to avoid positive semi-definite or negative definite updates due to rounding errors, i.e., to enhance numerical stability [33, 62]. Since our aim is different, i.e., to update J via a direct procedure, we derive a factorized secant update scheme (FSU) for B_+ of the Brodlie form [63]. The same secant condition should now hold for the $B_+ = (I + vy^T)B(I + vy^T)^T$ that is very similar to the one introduced by matrix $B = JJ^T$ and B is, by construction, positive definite. We will show that this FSU scheme reduces the total computational costs to $\sim \mathcal{O}(n^2)$ for each iteration, which remains tractable even for large n. To further reduce the requirements and avoid matrix storage, we cast FSU in a new recursive scheme inspired by limited-memory BFGS (L-BFGS). The L-BFGS method was earlier developed [64] to address large-scale problems, and has the advantage that the amount of storage (and thus the cost per iteration) can be controlled by the user, while retaining good overall performance. Our limited-memory FSU (L-FSU) scheme stores only three vectors of length n per iteration, and provides means to further restrict the computational costs per iteration by limiting the number of stored corrections (m) incorporated in J (and thus B). The analogy between the QN and S-QN frameworks is

update scheme minimization method	Full	Truncated
QN method	BFGS	L-BFGS
S-QN method	FSU	L-FSU

Table 3.1

illustrated in Table 3.1. This chapter is organized as follows: in the theory section, we derive the factorized FSU and L-FSU scheme, and quantify how further reduction of computational costs and storage is possible with L-FSU. In section 3.3 we consider the performance of FSU and L-FSU for a set of coupled harmonic oscillators. We consider this system for simple reasons: a) the Hessian H is analytically known and the convergence properties of B_k, as determined by FSU and L-FSU, can be quantitatively analyzed, b) multiple time and length scales play a role in the dynamics of this system, c) the system is a starting point for coarse-grained protein modeling. In the analysis we focus on the convergence of $B \rightarrow H^{-1}$, the presence of cooperative motion along the sampling pathway and the sampling distribution at long time scales. In particular, we focus on the effect of truncation (L-FSU) on these properties and the determination of a good history depth m. We will shortly elaborate on additional properties of the method, e.g., how the mobility can be used for introducing a multiplicity of time scales and the efficient and automated calculation of correlations in local or global minima.

3.2. Theory

3.2.1. S-QN

The S-QN method is based on a new stochastic Langevin equation for general n-dimensional potentials Φ given by [46]

$$\mathrm{d}\mathbf{x} = \left[-B(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T \nabla \cdot B(\mathbf{x})\right] \mathrm{d}t + \sqrt{2k_B T}\, J(\mathbf{x})\mathrm{d}W(t), \tag{3.6}$$

where $J(\mathbf{x})$ is related to the mobility $B(\mathbf{x})$ through

$$B(\mathbf{x}) = J(\mathbf{x})J(\mathbf{x})^T \ . \tag{3.7}$$

The new second term in the right hand side of equation (3.6) is the spurious drift term or flux caused by the random force. The crucial ingredient of our S-QN method is the mobility $B(\mathbf{x})$ or, in the discrete form, the $n \times n$ matrix B. We have previously discussed that our choice for the mobility matrix is inspired by Newton methods, i.e., H^{-1}, the inverse Hessian of Φ [46]. We relied on the BFGS standard in Quasi-Newton numerical optimization for constructing a series of positive definite matrices B_k (with k the time index in the discretized Langevin equations), such that $B_k \rightarrow H^{-1}$ under specific conditions [46]. These B_k constitute the adaptive compound mobility matrix that responds to the energy landscape by a memory function. We note that the (inverse) Hessian for the potentials Φ considered in the examples section is always a constant. Consequently, the spurious drift term in (3.6) is negligible for all B_k due to the closeness property and vanishes completely when B_k has converged to the inverse Hessian. For simplicity, we have therefore disregarded the spurious drift term in the remainder. We validated this explicitly for the systems considered in the examples section. The S-QN method and our new update algorithms for the mobility are, however, not in any way restricted to this special case. In particular, the update scheme for (3.6) in Appendix B.1 shows how the general case is only a correction to this special case, at the expense of additional costs.

We note that efficiency, i.e. avoiding the computation of the exact Hessian for large n, is not the only reason for the choice of BFGS. In general n-dimensional problems, the Hessian can and will become negative definite or even singular in parts of the energy landscape. This results in conditions for B that are principally equal to the implicit condition for the mass matrix in FA: B should *always* be positive definite to guarantee the existence of J in (3.7). When the secant condition is satisfied the BFGS update method guarantees the construction of such a positive definite B. Nevertheless, the sampling path will have to cross over concave and flat regions of the energy landscape, and we need to somehow adapt the mobility in these regions [46] (see in the body of this Chapter). The most important feature of the general methodology is that the mobility B always exists and remains positive definite, since the DFP update method constructs an *approximate* inverse Hessian, even when the Hessian itself is singular. This property is equivalent to the somewhat ad-hoc regularization in FA methods, but *automatic*. By using this non-singular approximation, sampling of the longer wavelength modes associated with zero (and other very small) eigenvalues of the Hessian is enhanced, hence the automatic scaling in our system. It is this property that will allow one to introduce a multiplicity of time steps by taking differential steps in different directions while maintaining thermodynamic consistency. Hence, the S-QN method bridges between general directed search methods such as QN and random search methods such as Monte Carlo (MC) or simulated annealing (SA). The QN method only ensures that the sampling path on this hypersurface is always in the descending direction, and is therefore principally *local*. In particular, QN does not sample according to a distribution. The update schemes in MC/SA incorporate global hypersurface information only weakly in the form of rules for acceptance or rejection based on a sampling distribution that favors configurations $\mathbf{x}$ with lower $\Phi(\mathbf{x})$ [20, 65]. A well-known drawback of this method is performance: for large systems (large n), the random sampling and subsequent substantial increase of the number of sampling points or function evaluations can make these algorithms computationally intractable.

3.2.2. The factorized secant update scheme

As mentioned in the introduction, a factorized update scheme was earlier developed to circumvent problems with the positive definiteness of B due to numerical errors [33, 62]. The method of Goldfarb updates a lower triangular matrix L to $\overline{J}$, followed by a decomposition of the matrix $\overline{J}$ into an orthogonal Q and a right triangular matrix R, i.e., a QR factorization, to obtain the new L with $\overline{J}\overline{J}^T = B^{-1}$. Since $B^{-1} = LQ^TQL^T = LL^T$, the next QN direction ($\Delta\mathbf{x}$) can be determined by solving $LL^T\Delta x = -\nabla\Phi$. Here, the particular reason for developing a factorized scheme for the QN-matrix $J(\mathbf{x})$ is rather different. In particular, apart from the drift term

$$-B(\mathbf{x})\nabla\Phi(\mathbf{x})\mathrm{d}t = -J(\mathbf{x})J(\mathbf{x})^T\nabla\Phi(\mathbf{x})\mathrm{d}t, \tag{3.8}$$

such a scheme enables a *direct* calculation of the noise term

$$\sqrt{2k_BT}\,J(\mathbf{x})\mathrm{d}W(t), \tag{3.9}$$

without an additional Cholesky factorization of the matrix B. In the following, we consider a discrete set of equations, with equidistant time steps labeled by k (starting from $t = kdt = 0$). In line with common practice in QN, the matrix B_k (the approximate of the inverse Hessian) is updated each iteration step to obtain a new matrix $B_{k+1} = B_k + \Delta B_k$, with ΔB_k a correction matrix. Suitable conditions for incorporating second-order information of Φ in this new matrix should be formulated, and we use the standard secant condition, based on expanding $\nabla\Phi$ in $\mathbf{x}_k - \mathbf{x}_{k+1}$ around the new point $\mathbf{x}_{k+1}$ as a Taylor expansion

$$\nabla\Phi(\mathbf{x}_k) \approx \nabla\Phi(\mathbf{x}_{k+1}) + \nabla^2\Phi(\mathbf{x}_{k+1})(\mathbf{x}_k - \mathbf{x}_{k+1}). \tag{3.10}$$

The property that B_{k+1}^{-1} approximates $\nabla^2\Phi(\mathbf{x}_{k+1})$ is equivalent to the secant condition given by

$$B_{k+1}\mathbf{y}_k = J_{k+1}J_{k+1}^T\mathbf{y}_k = \mathbf{s}_k, \tag{3.11}$$

where $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ and $\mathbf{y}_k = \nabla\Phi(\mathbf{x}_{k+1}) - \nabla\Phi(\mathbf{x}_k)$. Other properties that B_{k+1} should inherit from B_k include symmetry and positive definiteness. Unlike in standard QN these properties of B_{k+1} are automatic, since the product JJ^T is always symmetric positive definite for non-singular matrices $J \in \mathbb{R}^{n\times n}$. To determine a *unique* update J_{k+1}, we have to impose an additional condition on J. From all matrices J that satisfy the secant condition (3.11), we determine the one that is *closest* to J_k in some sense (see below), such that useful information stored in J_k is not lost in the update. The proximity condition, giving rise to a unique J_{k+1}, is casted into

$$\min_{J_{k+1}} \|J_{k+1} - J_k\| \tag{3.12}$$

$$J_{k+1}\mathbf{v}_k = \mathbf{s}_k \tag{3.13}$$

$$J_{k+1}^T\mathbf{y}_k = \mathbf{v}_k. \tag{3.14}$$

where the last two equations express the secant conditions on J_{k+1}. The convergence property of J is hereby satisfied: if all curvature information is stored in J_k, it is automatically inherited by $J_{k+1} = J_k$. There exist a nonsingular J_{k+1} satisfying (3.11), if and only if the curvature condition $\mathbf{s}_k^T\mathbf{y}_k > 0$ holds. When $\|\cdot\|$ is the Frobenius norm, the solution to (3.12) and (3.13) is even unique and J_{k+1} is given in terms of vectors $\mathbf{s}_k$ and $\mathbf{v}_k$. The uniqueness and existence proof is analog to the proof [33] given for

$$\min_{\bar{J}_{k+1}} \|\bar{J}_{k+1} - L_k\|, \tag{3.15}$$

where L_k is a lower triangular matrix, with the secant condition on $B_{k+1}^{-1} = \bar{J}_{k+1}\bar{J}_{k+1}^T$. We note that the lower triangular matrix L in this scheme was chosen for convenience: the relation for the update Δx, $LL^T\Delta x = -\nabla\Phi$, is easily solved for such L by forward and backward substitutions. Here, we want the update J_{k+1} in closed form (in terms of $\mathbf{s}_k$ and $\mathbf{v}_k$) instead, because this allows us to cast the scheme into a recursive form. In the next section we show how this recursive scheme can be exploited for limited-memory purposes. Using equation (3.14), $\mathbf{v}_k$ can be determined (see appendix B.2) and J_{k+1} is given by

$$J_{k+1} = J_k + \frac{\alpha_k\mathbf{s}_k\mathbf{y}_k^T J_k - \alpha_k^2 J_k J_k^T\mathbf{y}_k\mathbf{y}_k^T J_k}{\mathbf{y}_k^T\mathbf{s}_k}, \tag{3.16}$$

with

$$\alpha_k^2 = \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{y}_k^T J_k J_k^T \mathbf{y}_k}. \tag{3.17}$$

We can take the square root if $\mathbf{y}_k^T \mathbf{s}_k > 0$, the curvature condition, assuring the positive definiteness of the QN-matrix. In particular, we consider the positive root of α_k for which J_{k+1} is minimal in (3.12). We note that, in order to enhance the sampling of the potential energy hypersurface in concave or flat regions, we avoid the usual backtracking algorithms and use a constant increment dt instead. Gradient information is therefore always only calculated once during each update. It is essential to have a good strategy when the curvature condition is violated. In such case, we update $\mathbf{s}_k$ and $\mathbf{y}_k$ but keep the matrix B fixed, i.e., $J_{k+1} = J_k$, to ensure positive definiteness. As a trade-off, the secant condition $B_{k+1}\mathbf{y}_k = B_k\mathbf{y}_k = \mathbf{s}_k$ is not necessarily satisfied. However, since the next step is effectively a re-initialization of the scheme with $J_0 = J_k$, the secant condition is restored in step $k+2$ and can be disregarded at step $k+1$. We refer to our previous work [46] and the discussion section for more details on this particular choice.

We conclude that the factorized secant update (FSU) of (3.16) gives rise to a direct calculation of the drift (3.8) and noise (3.9) terms without further factorizations. Since the FSU update for B is equivalent to the update obtained from the Davidon-Fletcher-Powell (DFP) method of the convex BFGS family (see appendix B.2) we shortly review some general properties of DFP for a quadratic objective function such as Equation (3.21). Fletcher and Powell showed that, for a non-singular Hessian H, the matrix B converges to H^{-1} and x_k finds the minimum of Φ in exactly n steps if *exact* line searches are used [66]. The proof is based on showing that $\mathbf{s}_k$ ($k = 0, ..., n-1$), with step size α_k selected based on exact line searches, are linearly independent eigenvectors of $B_n H$ with eigenvalue unity (in other words, $\mathbf{s}_k$ $k = 0, ..., n-1$ is an basis for $\mathbb{R}^n$ and $B_n H = I$). In reality, exact line searches are seldom used. They are considered inefficient, as often very many costly Φ-evaluations are required for the determination of the QN step size α_k in each iteration step. The preferred backtracking algorithms [21] are less efficient in terms of convergence, but the number of Φ-evaluations per iteration is also considerably reduced. In our procedure backtracking was not considered for various reasons, and the number of Φ-evaluations per iteration step is further reduced (to one). Returning to exact line searches [66], the update $B_{k+1} = B_k + A_k + A_k'$ consists of two rank-one contributions to B_k. The first contribution A_k ensures that the secant condition $B_{k+1}\mathbf{y}_k = \mathbf{s}_k$ is always satisfied, or, alternatively, that $\mathbf{s}_k$ is an eigenvector of $B_{k+1}H$ with unit eigenvalue. The second contribution A_k' deals with the convergence of B to H^{-1}, and one can show that $H^{-1} = \sum_{k=0}^{n-1} A_k'$. For an inexact or constant step size the vector $\mathbf{s}_k$ is in general not orthogonal to $g_{k+1} = \nabla\Phi(x_{k+1})$ and, consequently, the build-up of information of H^{-1} in A_k' will be (much) slower. We note that the noise in Equation (3.20) adds a 'random' displacement $\mathbf{d}$ to the QN-update, and this $\mathbf{d}$ is likely to also contain displacements orthogonal to g_{k+1}. For general inexact line searches, the update can be seen as successive rank reduction and rank restoration [67]. The first step A_k always reduces the rank of B_k by one. The second step A_k' restores the rank to the rank of B_k and gives rise to a positive-definite B_{k+1}, all provided that the curvature condition $\mathbf{y}_k^T\mathbf{s}_k > 0$ is satisfied. The difference $B_{k+1} - B_k$ is a symmetric matrix of rank at most two, with column and row spaces spanned by $\mathbf{s}_k$ and $\mathbf{h}_k = B_k\mathbf{y}_k$. A rank-two update is obtained only for linearly independent $\mathbf{s}_k$ and $\mathbf{h}_k$; otherwise the update is of rank one [68]. Davidon [69] showed that the generalized eigenvalue problem $B_{k+1}\mathbf{z} = \lambda B_k\mathbf{z}$ has $n-2$ unity eigenvalues and 2 eigenvalues that may differ from 1. These eigenvalues can be determined analytically as a function of $a = \mathbf{y}_k^T B_k\mathbf{y}_k$, $b = \mathbf{y}_k^T\mathbf{s}_k$ and $c = \mathbf{s}_k^T B_k^{-1}\mathbf{s}_k$ [68, 69].

However, when minimizing a system of size n, the computational and storage costs may still be impractical for large n. In the next section we therefore propose an efficient implementation of FSU, which leads to a reduction of the n^2 storage and $\mathcal{O}$ (n^2) manipulations of the scheme described in Equation (3.16).

3.2.3. Recursive limited-memory update

Limited memory methods in the Broyden family are in general based on truncating the history in the iterative update schemes for the approximate Hessian or inverse Hessian B. The direct advantage is in storage: at iteration k only a fixed number m (the history depth) of vector sets is stored, instead of the linearly growing number of sets ($\sim k$) in the FSU update scheme. In particular, for $k > m$ the oldest information contained in B is discarded and replaced by the new one. Although one could intuitively expect this truncation to affect the performance of the method, numerical evaluations show that this procedure for BFGS (the L-BFGS method) gives rise to good convergence properties in practice, even for small m [64]. A theoretical understanding of this property for general cases is still lacking.

In appendix B.4, we provide the details of a recursive scheme that is suited for both FSU and limited-memory FSU (L-FSU). The strategy is to avoid the use of expensive matrix-vector products by loop unrolling. Instead of n^2 storage for the matrix J

in Equation(3.16), this scheme requires storage of three vectors $\{\mathbf{s}_k, \mathbf{y}_k, \mathbf{h}_k\}$, each of length n. For L-FSU, this results in a total storage of $3mn$, with m in general small (see appendices). The starting matrix for the limited-memory updates, J_0, can be freely chosen, but should reflect the discarded information for the problem at hand. A standard choice is the same initial value for J_0 as considered for FSU, i.e., $J_0 = I$, but scaling $J_0 = \gamma_k I$ with

$$\gamma_k = \mathbf{y}_k^T \mathbf{s}_k / ||\mathbf{y}_k||^2 \tag{3.18}$$

was identified a simple and effective way of introducing a scale in the algorithm and improving its performance [64]. However, the analysis and the numerical examples (see next section) use $J_0 = I$. Just like in the L-BFGS the starting matrix J_0 can be freely adjusted in our scheme during the iterative process.

In contrast to FSU, one should take special care when the update is temporarily stalled due to a violation of the curvature conditions $\mathbf{y}_k^T \mathbf{s}_k > 0$. The limited-memory algorithm employs a shifting window of m vector-triplets $\{\mathbf{y}, \mathbf{s}, \mathbf{h}\}$ to update the B matrix at each k step, starting from the initial condition $J_0 = I$ (all eigenvalues = 1). Since the update scheme is of rank two at most, a maximum of $2m$ eigenvalues of B will deviate from $\lambda = 1$ (see also previous section) at *any* stage of the simulation. All local-curvature information, or, alternatively, information about the different modes in the system, is contained in the eigenvectors with eigenvalue $\lambda \neq 1$. The history contained in this window may stretch over a much longer range than simply anticipated from the recursive schemes, since B is not always updated. For the sake of the argument, we suppose $k = \bar{k} \geq m$ when the curvature condition is first violated and that the second violation takes place after an additional number of steps $\geq m$. The general case is more involved but straightforward. In appendix B.3 we have shown that, in order to satisfy the secant condition at all times, adapted $\underline{V}_k$ based on $\tilde{\mathbf{h}}$ instead of $\mathbf{h}$ are required after m updates. Upon updating B at step $\bar{k} + 1$, one option is to disregard the information contained in B and restart the L-FSU scheme with $J_0 = I$, since by definition this information can be restored in m steps. As in FSU (see section 3.2.2) one can also see the next step as an effective re-initialization of the scheme with $J_0 = J_{\bar{k}}$, based on the logic that especially for large m the matrix $J_{\bar{k}}$ contains valuable Hessian information. In the latter case, one can either store the $n \times n$ matrix $J_{\bar{k}}$ or build $J_{\bar{k}}$ recursively from its particular window of past vector-triplets (a maximum of m). In both cases, the use of $J_{\bar{k}}$ introduces additional memory and/or computational requirements. The secant condition is restored in step $\bar{k} + 2$ and can be disregarded at step $\bar{k} + 1$. Whatever restart method is most efficient depends on the values of m and n. For both J_0, however, it is important to note that the L-FSU scheme should not truncate during the first m step, i.e., L-FSU and FSU after restart are equal and V_k based on $\mathbf{h}$ should be considered in the update.

We shortly analyze the computational load by considering the total number of elementary operations for one update cycle equation (3.6). As a reference, the FSU update (see equation (3.16)) requires $4n^2$ multiplications (first calculate $J_k^T \mathbf{y}_k$ and reuse this vector in the update). Calculating the drift part directly from $J_k J_k^T \nabla\Phi$ requires $2n^2$ multiplications. A total of n^2 multiplications are needed for the noise part $J_k \mathrm{d}W$. Summing all up, the total number of multiplications using the FSU update is $7n^2$. In the recursive L-FSU update scheme, $10mn + 2n$ and $2mn + n$ multiplications are required for the calculation of the drift and noise term, respectively (see appendix B.4 for details). A total of $12mn + 3n$ multiplications is therefore required for the L-FSU update. A conclusion from this simple analysis is that L-FSU is more efficient than the FSU update scheme, in terms of computational as well as memory costs for $m \leq n/2$ and definitely better than using Cholesky decomposition. Since memory and computational constraints are typically problematic for large n, this is a much desired property.

3.3. Results and discussion

In the following examples, we have discretized equation (3.6) into

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}_k, \tag{3.19}$$

$$\Delta\mathbf{x}_k = -J(\mathbf{x}_k)J(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\Delta t + \sqrt{2k_B T}\, J(\mathbf{x}_k)\Delta W. \tag{3.20}$$

using an explicit Euler scheme. Here $\Phi = \Phi_{\text{spring}}$ is a harmonic potential, describing the behavior of n connected particles at $\mathbf{x} = (x_1, ..., x_n)^T$ on a line, with

$$\Phi_{\text{spring}} = \sum_{i=1}^{n-1} (x_{i,i+1} - 1)^2; \tag{3.21}$$

and $x_{i,i+1} = |x_i - x_{i+1}|$ the distance between particle i and $i+1$. For simplicity, we have set the equilibrium distance and spring constants to unity. We note that the considered model is related to the well-known Rouse model in polymer dynamics, which we can obtain by taking a spring constant $3k_B T b^{-2}$, with b the Kuhn segment length, and a vanishing equilibrium length in equation (3.21) and constant mobility $B = JJ^T = \zeta^{-1}$ in equation (3.20), where ζ is the friction coefficient due to the surrounding medium.

Although our one-dimensional (1D) particle-spring system seems trivial at first sight, it possesses many features, in particular critical slowing down, that are present in non-trivial models, which automatically require simulation. An clear advantage is that the analytic Hessian H is easily calculated, and H can thus be directly compared to B_k^{-1}. In particular, this Hessian $H_{\Phi_{\text{spring}}}$ is a tridiagonal matrix,

$$H = \begin{bmatrix} 2 & -2 & 0 & & \\ -2 & 4 & -2 & & \\ & \ddots & \ddots & \ddots & \\ & & -2 & 4 & -2 \\ & & 0 & -2 & 2 \end{bmatrix}, \tag{3.22}$$

which is singular. The kernel or null-space of H, Null(H), is spanned by the n-dimensional vector $\mathbf{1} = [1 \ldots 1]^T$, corresponding to the translational invariance of the energy potential Φ_{spring}, i.e., the insensitivity of the potential to a translation of the string of particles as a whole. An equivalent Gaussian model was analytically considered by Dünweg [47], who derived it from the well-known Landau-Ginzburg Hamiltonian by omitting the ϕ^4 term. Dünweg showed that the correlation times of longer-wavelength modes (smaller p) become increasingly large. With an increasing length-scale, the system therefore becomes increasingly soft or - correspondingly - increasingly sluggish.

We concentrate on three features: 1) the similarity between B_k and the inverse Hessian, 2) the sampling performance, and 3) the sampling distribution.

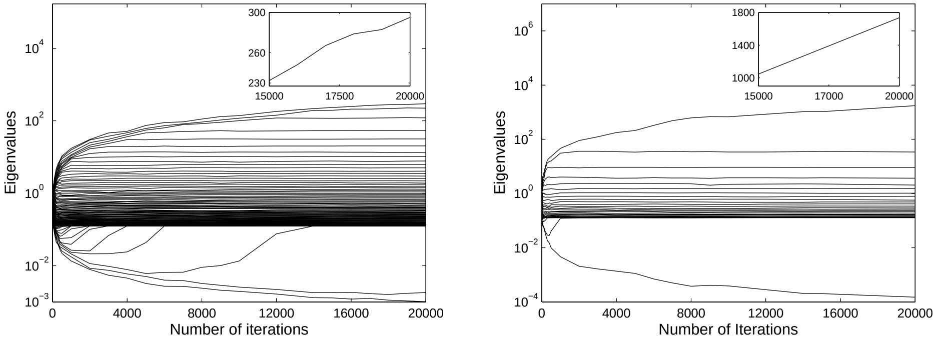
3.3.1. Comparison of the adaptive mobility to the inverse Hessian

One consequence of the singularity of the Hessian is that a direct comparison between H^{-1} and B_k is impossible. We can use the *generalized* inverse matrix H^- [70], that exists for any H such that $HH^-H = H$, but for a singular matrix it will not be unique. In particular, for any given generalized inverse H^-, the class can be generated by $H^- + U - H^-HUHH^-$ with U any arbitrary matrix. For a non-singular H the inverse is unique and $H^- = H^{-1}$. The adaptive mobility B_k itself is positive-definite by construction, and the inverse B_k^{-1} can be obtained in $O(n^3)$ operations. However, we circumvent the additional inversion at each iteration step k, and focus on the properties of B_k itself, starting with the eigenvalues of B_k. Rates of convergence can be determined from $||HB_kH - H||_{\text{F}}$, the Frobenius norm, as B_k itself may converge to any member of the class of generalized inverses of H. In the discussion we show that for the considered system it is in principle possible to use $B_kH - I$ instead. In addition, we deal with the singularity of H by constraining the system. The first and rather crude option is to reject displacements associated with Null(H) in the update of B_k, by applying an affine translation after each step such that the chain's center of mass $c_{k+1} = c(\mathbf{x}_{k+1}) = \sum_{i=1}^{N} x_i$ is always reset to the initial value c_0. A better option, that resolves the singularity itself, is to regularize the system by adding a penalty function containing $c(\mathbf{x})$ to the harmonic potential in equation(3.21). First, we evaluate the properties of B_k for the unconstrained case. Consequently, we relate the eigenvalue spectrum for this B_k to the eigenvalue spectrum of the regularized H, using elementary linear algebra.

For a quadratic potential, the build-up of information in B_k is *independent* of the variables Δt and k_BT in equation (3.20), as long as B_k is updated ($\mathbf{y}_k^T\mathbf{s}_k > 0$) for each k. Moreover, this process is insensitive to $\mathbf{x}_0$, the initial state in equation (3.20). For these reasons, the spectral properties and rates of convergence of B_k considered in the next subsections are universal. The sampling pathway itself depends on the values of k_BT and Δt, as well as the initial state, and should be chosen with care. In particular, we have selected $k_BT = 0.01$ in Figures 3.1-3.5. For the results shown in Figure 3.1 and 3.3 $\Delta t = 0.0001$ and $\Delta t = 0.01$ for Figure 3.2 . In Figures 3.4 and 3.5 we chose $\Delta t = 10^{-6}$ since a small time step promotes $\mathbf{y}_k^T\mathbf{s}_k > 0$.

3.3.1a. FSU

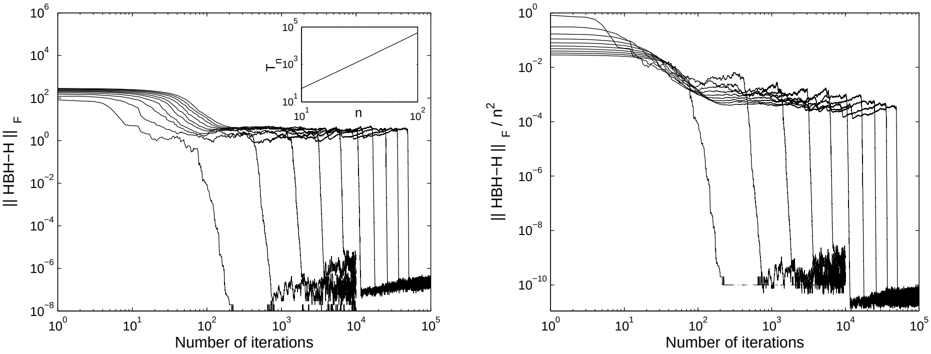
We focus on the convergence of B_k, obtained by FSU, for the spring potential (3.21) using (3.20). We first consider the eigenvalues of the mobility matrices B_k (we omit k dependence for simplicity of notation). Multiple length and time scales should play an important role in systems with a larger number of degrees of freedom, and we start with $n = 100$. The simulated k-evolution of the eigenvalue spectrum of B is shown in Figure 3.1a. It is clear that the eigenvalues, apart from the extremes, approach a constant value with increasing k within the limited time of simulation (20000 steps). The largest (smallest) eigenvalues continue to increase (decrease) with increasing k: focusing on the maximum $\lambda_{\max}$ this increase is approximately linear with k. Visual inspection of the spring dynamics (see next section for details) shows a dominant and random movement of the spring as a whole at later k stages, signaling that the smallest eigenvalue of the *inverse* of the non-singular B indeed converges to zero with increasing k (or alternatively, $\lambda_{\max} \to \infty$ for B). Force contributions along the eigenvector $\mathbf{1}$, the null-space of H, will thus be amplified with increasing k, leading to the dominance of coordinated but diffusive movement of the string at later stages. Figure 3.1b shows the eigenvalue spectrum for a system with a smaller number of degrees of freedom ($n = 27$) and is qualitatively very similar. From a comparison between both spectra we observe that the time scale at which most eigenvalues approach constant values depends on the number of degrees of freedom n. We quantify the simulated rate of convergence further by considering the evolution of $||HBH - H||_{\mathrm{F}}$ (Figure 3.2a) and the scaled norm $(1/n^2)||HBH - H||_{\mathrm{F}}$ (Figure 3.2b) for varying $n = 10, 20, ..., 100$. Since the Frobenius norm sums over all elements of the $n \times n$ residual matrix $HBH - H$, the scaled norm in Figure 3.2b provides the average residual per matrix element. From Figure 3.2, we observe that for all considered n the matrix B_k converges to a generalized inverse H^-, within certain error bounds, and that the residual norm has a typical S-shape. In the initial stages, the FSU scheme starts updating $B_0 = J_0 = I$ (all eigenvalues = 1) if and only if the curvature condition is satisfied. Focusing on selected simulations shown in Figure 3.2, we indeed observed that either one or two eigenvalues become $\neq 1$ after each update. In addition to these two eigenvalues, all other eigenvalues $\neq 1$ are adjusted in each update. All eigenvalues are therefore $\neq 1$ for $k \in [n/2, .., n]$ steps. From Figure 3.2b, we can identify this process as the first smooth and slowly decreasing part of the S-curve that terminates in a plateau value after approximately n steps. The second stage, the noisy plateau region, deals with correction and further build-up of linear independent



(a) *Spectrum vs iteration index k for n* = 100 **(b)** *Spectrum vs iteration index k for n* = 27

Figure 3.1: *Eigenvalue spectrum of the mobility matrices constructed by FSU for the spring potential. The insets in (a) and (b) display the evolution of the largest eigenvalue.*

curvature information via new $\mathbf{s}_k$ and $\mathbf{h}_k$. The plateau value of the residual norm itself seems independent of n (Figure 3.2a). This plateau could be seen as an exited state with a lifetime T_n that depends on the dimensionality n of the system. From the T_n vs n plot, shown as insets on a log-log scale in Figure 3.2a , we determined the life time as $T_n = 0.05n^3$. The noisy plateau region terminates via a drop, signaling that the construction of a (generalized) inverse Hessian H^- becomes completed. The constant slope associated with this drop was used in the determination of T_n. Combining



(a) $||HB_kH - H||_\mathrm{F}$ *vs iteration index k for various n*

(b) $\frac{1}{n^2}||HB_kH - H||_\mathrm{F}$ *vs iteration index k for various n*

Figure 3.2: *Rate of convergence of the mobility matrix* B_k *to the generalized inverse Hessian* H^{-1} *for* $n = 10, 20, 30, ..., 100$. *For all cases the initial guess* $B_0 = I$*, as a result the residual norm at* $k = 0$ *increases with n. The inset in Figure 3.2a shows the plateau length* T_n *vs n in a log-log scale.*

Figures 3.1 and 3.2 for $n = 27$, we find that prior to the drop most intermediate eigenvalues have almost reached their final and constant value (Figure 3.1). The drop at $k \approx 1000$ (Figure 3.2) corresponds to the convergence of all $n-2$ eigenvalues to constant values, although small fluctuations around these values can be observed at later stages. Only the two extreme eigenvalues are still increasing (λ_max) and decreasing (λ_min) for $k > 1000$. The fact that λ_min and λ_max do not level off with increasing k shows that the condition number of B_k increases with k and is consistent with the rank-two nature of the update scheme. Moreover, these eigenvalues are inversely proportional to the extremes for H, corresponding to the fastest and slowest modes in the system. This observation is consistent with the simple analysis in the discussion section, which anticipated that the relaxation rates associated with the slowest (fastest) modes is increased (decreased) with respect to standard Langevin dynamics ($B_k = I$), respectively. For $n = 100$ the instantaneous drop occurs at $k \approx 50000$ (Figure 3.2). We conclude that the earlier observation, i.e., all $n-2$ eigenvalues have converged at $k = 20000$, is not strictly valid. Close re-examination of Figure 3.1 shows indeed that the one but lowest eigenvalue has apparently not converged at $k = 20000$. Also for $n = 100$ most intermediate eigenvalues have (almost) reached their steady state value for rather small k.

3.3.1b. Regularization

A non-singular $\tilde{H}$ is obtained for the constrained potential given by

$$\tilde{\Phi} = \Phi + \Phi_{\text{com}}, \tag{3.23}$$

with $\Phi_{\text{com}} = (c_k - c_0)^2$ and $c_k = c(\mathbf{x}_k) = \sum_{i=1}^{N} x_i$. Adding this term to the potential has the effect of conditioning the eigenvalues. We focus on the smallest system from the previous section, $n = 27$, for illustration purposes. The eigenvalues of $\tilde{B}$ obtained by FSU (not shown here) level off to constant limiting values for increasing k ($k > 1000$), including λ_{max} that is now bounded. Owing to the non-singular $\tilde{H}$, we can directly compare the steady-state eigenvalue spectrum of $\tilde{B}$ to the analytic spectrum for $\tilde{H}^{-1}$. In addition, we can compare this analytic spectrum to the steady-state eigenvalue spectrum of B itself (see Figure 3.1b). Linear algebra provides the tools for such a direct comparison, in terms of an expression for the change in eigenvalues of a matrix when a rank one matrix is added. Let $\mathbf{u}$ and $\mathbf{v}$ be two n-dimensional vectors and $\mathbf{u}$ an eigenvector of the $n \times n$ matrix A. The eigenvalues of $A + \mathbf{u}\mathbf{v}^T$ are then given by:

$$\sigma(A + \mathbf{u}\mathbf{v}^T) = \{\sigma(A) \setminus \lambda_{\mathbf{u}}\} \cup \{\lambda_{\mathbf{u}} + \mathbf{u}^T\mathbf{v}\}, \tag{3.24}$$

where $\lambda_{\mathbf{u}}$ is the eigenvalue from corresponding eigenvector $\mathbf{u}$. Taking $A = B^{-1}$, $\mathbf{u} = [1 \dots 1]^T$ and $\mathbf{v}^T = [2 \dots 2]$, and thus $\mathbf{u}\mathbf{v}^T$ equal to the Hessian of Φ_{com} (H_{com}), we can directly compare the spectrum of $B^{-1} + H_{\text{com}}$ to the analytic spectrum of $\tilde{H}$. For consistency, we plot in Figure 3.3 the inverse of the eigenvalues calculated using this relation. We note for completeness that the eigenvalue $\lambda_{\mathbf{u}}$ in equation (3.24) is the previously mentioned $\lambda_{\text{max}}^{-1}$ of B. From the spectra in Figure 3.3, we observe that the smallest eigenvalues of B (+, $k = 10000$) and $\tilde{B}$ (○, $k = 2000$) coincide, while the largest eigenvalues deviate considerably. The eight largest eigenvalues of B are increasingly exceeding the ones for $\tilde{B}$, and we note that the highest eigenvalue ($\lambda_{max} \approx 351$) of B even exceeds the chosen vertical axis limit. Conditioning the eigenvalue spectrum by the penalty function Φ_{com} in equation (3.23) thus only acts on

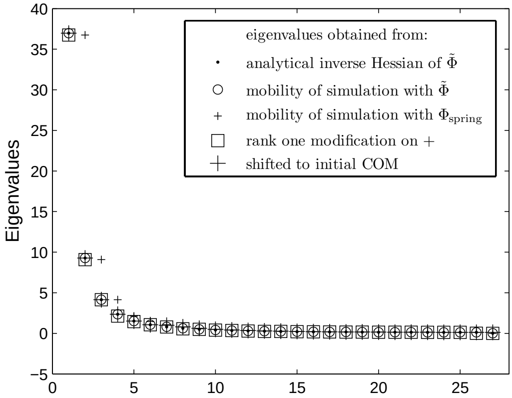
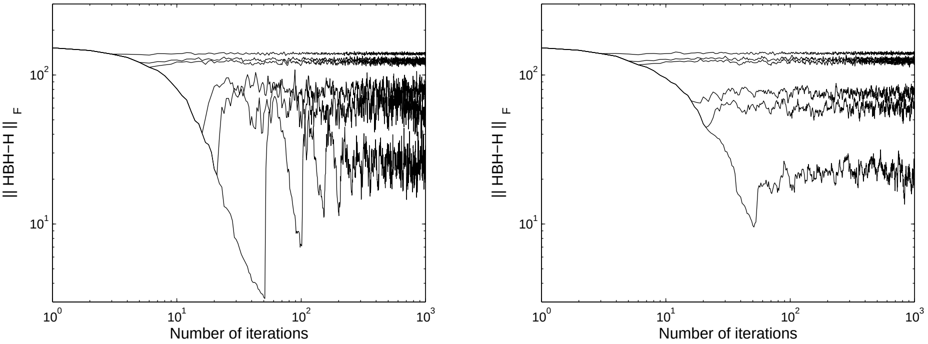


Figure 3.3: *Comparison of the eigenvalues of the analytic inverse Hessian and the mobility matrix as constructed by FSU for the constrained* $\tilde{\Phi}$ *and* Φ_{spring}.

the smallest eigenvalue of $H \sim B^{-1}$, as could be expected. Comparing the spectrum of $\tilde{B}$ ($\circ$) and the analytic spectrum for $\tilde{H}^{-1}$ ($\cdot$) shows that they coincide in detail. We conclude that for $\tilde{\Phi}$, FSU constructs a perfect approximate of the inverse Hessian starting from $k \approx 1000$. Finally, we used the relation (3.24) to project the spectrum of B onto the analytic values for $\tilde{H}^{-1}$ ($\square$). From the perfect match, we conclude again that B^{-1} is an accurate approximate of H, in particular at the considered later k-stages ($k = 10000$). Finally we carried out an additional FSU simulation for the unconstrained potential Φ. After each k update, the chain center of mass is reset to the original position by an affine translation. The eigenvalue spectrum (denoted by large +) coincides with the analytic values, and we conclude that even with this ad-hoc regularization procedure FSU generates a very accurate inverse Hessian.

3.3.2. L-FSU

From numerical evaluation on a range of test problems, L-BFGS is known to be an efficient general-purpose method for determining optimal solutions in nonlinear problems, i.e., $\mathbf{x}'$ such that $\Phi(\mathbf{x}')$ is optimal, even for relatively small m [64]. L-BFGS is therefore the most popular member of the (convex) Broyden family of limited-memory methods, but the performance of L-DFP is known to be comparable [71]. As far as we know, some properties, like the convergence of B_k to H^{-1}, have not been considered in detail. The convergence to optimal solutions suggests that at least some Hessian information is stored in B. The important issue considered here is which, and how much, Hessian information is stored in B, and which m is optimal in this sense. We consider L-FSU for $n = 30$. The history depth m is varied from small $(2, 4$ and $5)$ to larger values $(15, 20$ and $50)$. Figure 3.4a shows the (unscaled) residual norm as a function of k and m. For all m, the norm in the initial m steps is equal to the FSU norm (see Figure 3.2). Starting from $k = m$, older information is disregarded in favor of new information. For small m $(m = 2, 4, 5)$, the residual norm is observed to oscillate around a constant value that is roughly equal to the residual norm of B_m. The effect of truncation is more distinct for larger m and gives rise to large oscillations that damp out while the residual norm reaches an almost constant value. The apparent periodicity of these oscillations shows an intricate effect. Since $n = 30$, all eigenvectors/eigenvalues are affected by the update process for $m > 15$ and H^{-} will be increasingly approximated by B_m. As shown before in Figure 3.2, the initial stages $(k < n/2)$ give rise to a fast decrease of the residual norm relative to the correction at later stages, which is associated with the noisy plateau region. Disregarding the V_k of these initial stages will naturally give rise to a jump in the



(a) *$||HB_kH - H||_F$ vs iteration index k, where B_k is obtained by L-FSU for $m = 2, 4, 5, 15, 20$ and 50.*

(b) *$||HB_kH - H||_F$ vs iteration index k, where B_k is obtained by L-BFGS for $m = 2, 4, 5, 15, 20$ and 50.*

Figure 3.4: *Comparison of the rate of convergence of the mobility matrix obtained by L-FSU and L-BFGS with various mto the generalized inverse for $n = 30$ and the spring potential. The first m updates updates of B_k in L-FSU and FSU coincide and the deflection point is decreasing with increasing m.*

residual norm. This information will be restored by the iterative process and give rise to apparent periodicity. As the information that can be incorporated in B_k is limited to m vector-triplets, also this residual norm will converge to a constant value, but this value may be slightly higher than the residual norm of B_m if the process of re-iterating the disregarded information cannot keep up with the truncation process. For this constant Hessian, one could therefore choose to keep $B_k = B_m$ fixed during the iterative process for $k > m$. For general potentials, the Hessian will vary and the B_k obtained by L-FSU will be slaved by the pathway on the energy hypersurface. The k evolution of the eigenvalues of B_k is shown in Figure 3.5, for varying m. Except for the eigenvalues = 1, originating from $J_0 = I$, none of the eigenvalues levels off to a constant value. Instead, and although hidden by the logarithmic scale used in Figure 3.5, they show tiny oscillations around a constant value due to the truncation process. It shows the convergence of B_k to a stationary state $\overline{B}(m)$. It is clear that $2m$ eigenvalues differ from unity and that the range of the eigenvalue spectrum increases with increasing m. Since these eigenvalues directly correspond to a *multiplicity* of time steps, we conclude that the multiscale nature is enhanced by considering larger m.

Finally, we validate the new L-FSU method by considering L-BFGS for the same system and varying m (see Figure 3.4b). We used Cholesky decomposition for the noise factor. We identify the same type of oscillation around a constant value as in L-FSU for small m (Figure 3.4a). Apparently, the L-BFGS scheme is more stable against truncation for large m by the particular form of the update scheme. It only shows that the update schemes are conceptually different. From Figure 3.4 and $k < m$, one can conclude that FSU gives a better approximation for the inverse Hessian than the standard BFGS for this particular potential. Nevertheless, the residual norms of the stationary states $\overline{B}(m)$ are roughly equal to the ones obtained from L-FSU.

3.3.3. Multiscale simulation

Although some information is provided in the previous sections, we further concentrate on the multiscale nature of FSU and L-FSU. Analogous to effective diffusion, the mobility on larger length and time scales (coordinated movements) can be determined from the root mean square displacement of the chain's center of mass. Throughout the rest of this chapter, the simulation variables are set to $\Delta t = 0.01$ and $k_B T = 10^{-5}$. We have chosen $n = 27$, in order to complement the results of the previous section. A relatively small n allows us to concentrate on the role of the adaptive mobility when one could assume that larger modes are less important. At the end of this section,

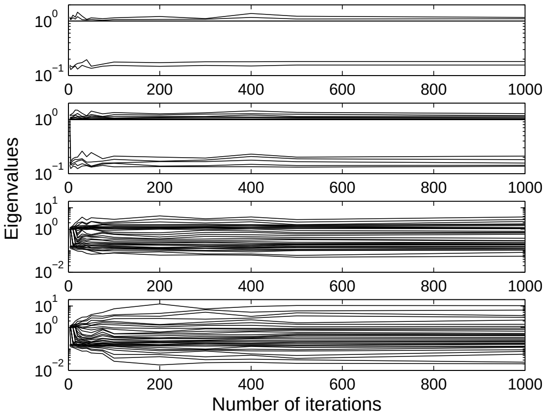
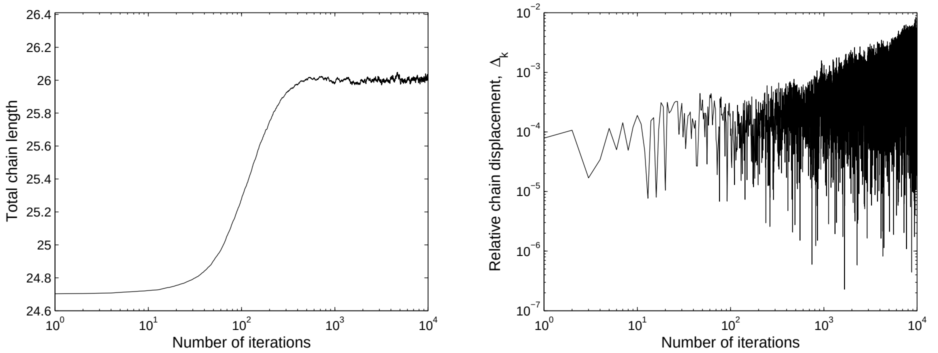


Figure 3.5: *Eigenvalues of B_k vs iteration index k for $n = 30$, where B_k is obtained by L-FSU. From top to bottom $m = 2, 4, 20$ and 50.*

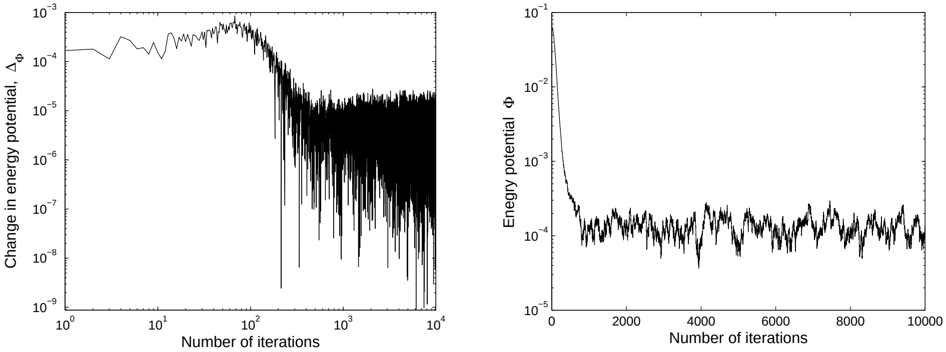
we consider the performance for $n = 100$ as well. Another important choice for the sampling behavior is the initial state $\mathbf{x}_0$. For $\mathbf{x}_0$ close (far) from the optimal state $\mathbf{x}'$ of Φ, with $\Phi(\mathbf{x}') = 0$, cooperative motion will play a less (more) significant role. We consider two different initial states. In the first scenario, we initially position all particles at a distance $x_{i,i+1} = 0.95$, rather close to the equilibrium distance. We note that the topography of the potential energy hypersurface around the optimal state is symmetric, since all spring constants were set to unity.

Starting with FSU, the chain's contour length and the relative chain displacement $\Delta_k = c_k - c_{k-1}$ are shown in Figure 3.6a and 3.6b, respectively. The potential energy difference between successive steps $\Delta\Phi_k = \Phi(\mathbf{x}_k) - \Phi(\mathbf{x}_{k-1})$ and the potential energy value are shown in Figure 3.6c and 3.6d. From Figures 3.1 and 3.2, we concluded that B_k is increasingly approximating H^-, while the largest eigenvalue continues to rise approximately linear with k. Figure 3.6a shows that the chain's contour length l rises to the equilibrium length $l_0 = 26$ very fast due to larger modes in the system. The line intersects with l_0 after ≈ 500 steps and fluctuates around l_0 at later stages. From Figure 3.6c and 3.6d one can see that the system has to overcome a small barrier during the equilibration process and that the potential energy hardly changes after the equilibrium contour length is reached. At $k \approx 1000$, the optimal state $\mathbf{x}'$ is reached and equation (3.20) starts sampling the basin around $\mathbf{x}'$ with a decreased and rather small noise amplitude. Apparently, the contour length l_0 is reached before all particles arrive at their equilibrium distances. From Figure 3.6b shows that Δ_k changes very rapidly between successive steps and that the average Δ_k increases monotonically with k. At later stages, Δ_k becomes much larger than Δt. We conclude from Figure 3.6 that the chain moves as a whole during the later stages, due to the increasing λ_{max} associated with displacements in the null-space of H. This chain movement is random, the true hallmark of diffusion. We also compared the performance of (3.20), with J_k determined by FSU, and conventional Langevin dynamics (CLD). The value of the scalar mobility in $B = MI$, resulting from friction due to the surrounding medium, depends to a high degree on the unknown topography of the energy hypersurface. This M constitutes an effective scaling of the time step Δt ($M\Delta t$ for all modes), which is automatic and mode-dependent in Langevin dynamics with curvature-dependent mobility. For the special case considered here, this topography is quite simple since all spring constants = 1, and $M = ||\tilde{H}^{-1}||_{\mathrm{F}}/||I||_{\mathrm{F}} = 1/n||\tilde{H}^{-1}||_{\mathrm{F}} = 7.4068$ would be a good value, but one need to explicitly include information of the analytic inverse Hessian for the constrained Φ. In general, and even for a Harmonic potential with different spring constants, the determination of an optimal M is much more difficult. In those cases M is determined on physical grounds or by trial simulations in the vicinity of the starting state. The starting point for FSU is $B_0 = I$, and we therefore consider $M = 1$ here. All motion is local and the noise samples around the steepest descent direction, i.e., the drift term for $B_k = I$. From Figure 3.7a we find that the contour length l approaches the equilibrium value l_0 but has not reached it at $k = 10^4$. The relative chain displacement Δ_k and potential energy difference between successive steps $\Delta\Phi_k$ for CLD are shown in Figure 3.7b and 3.7c, respectively. We find that $\Delta_k << \Delta t$ and the average Δ_k is apparently a very small constant. In other words, the chain center of mass moves only marginally due to displacements of individual particles and cooperative displacements are absent. The barrier of Figure 3.6c is absent in Figure 3.7c, signaling that also the pathway on the potential energy hypersurface is different. The potential energy Φ has almost reached the optimal $\mathbf{x}'$ at $k = 10^4$, but the amplitude of the noise is roughly equal during the whole simulation pathway. For L-FSU, we considered the sampling performance for $m = 5$ and $m = 50$. Figure 3.8a shows the contour length l, relative chain displacement Δ_k (Figure 3.8b) and potential energy difference between successive steps $\Delta\Phi_k$ (Figure 3.8c). It is clear that for small $m = 5$ the sampling performance is considerably reduced compared to FSU. The convergence of l and $\mathbf{x}_k$ to l_0 and $\mathbf{x}'$ respectively are comparable with CLD (Figures 3.7 and 3.8). From the analysis of eigenvalues (Figure 3.5, for $m = 4$) it is clear that the spectrum is very narrow and that the eigenvalues only slightly deviate from unity. In certain parts of the potential hypersurface, the direction of search will therefore be rather similar to the one in steepest descent, and some modes will be accelerated while others will be slightly damped. This effect can also be observed in Δ_k (Figure 3.8b). The spread of Δ_k marginally increases with k and the noise amplitude in $\Delta\Phi_k$ decreases only very mildly while approaching the optimal state of the system (Figure 3.8c and 3.8d). For larger $m = 50$, the performance improves. The equilibrium length l_0 is reached at $k \approx 2600$ and l fluctuates around l_0 at later stages. The optimal state is first found at roughly the same $k \approx 2500$, after which the basin is sampled. The contribution of larger modes in the displacement is reduced compared to FSU due to truncation (compare Figure 3.1a and Figure 3.5 for $m = 50$). Although not very significant, the spread of Δ_k increases with increasing k and the maximum displacement is larger than for $m = 5$ (Figure 3.8b). From Figure 3.8c, one can observe that also for $m = 50$ the simulation pathway crosses a small barrier. Note again the reduced noise amplitude close to the optimal state. We shortly consider the sampling performance for $\mathbf{x}_0$ further from the optimal $\mathbf{x}'$, where $x_{i,i+1}$ is randomly chosen between 0.5 and 5. In Figure 3.9a the contour length l and relative chain displacement Δ_k (Figure 3.9b) are compared for J_k derived by FSU, $J = I$ (CLD) and J_k derived by L-FSU for $m = 5$ and $m = 15$. In Figure 3.9a, l can be seen to drop very fast toward l_0 for both FSU and L-FSU, within $\approx$ 500 steps, and rather independent of m. In the case of L-FSU, the incorporation of curvature information speeds up this process considerably (compare to CLD) even for small m. When the chain has (almost) contracted to the equilibrium contour length l_0, cooperative chain movements in the drift term become less important, with an exception for the movement of the whole chain. The noise term in equation (3.20) gains importance since $\mathbf{x}_k$ is close to the optimal state $\mathbf{x}'$ at $k \approx 500$ in both schemes. As the noise contribution is fairly small ($k_B T = 10^{-5}$), the result is a distinct kink in the l-curve. We find that the overall convergence to the optimal state $\mathbf{x}'$ in (L-)FSU is one order of magnitude faster than for CLD. The information in Figure 3.9b supports this analysis. The Δ_k for FSU is again governed by the increasing λ_{max} or movements of the whole chain. For L-FSU and small m, full-chain movements are considerably damped due to the truncation process. We observe rather large Δ_k in the initial stages for both values of m due to cooperative motion in the drift term. At later stages, when for L-FSU the contribution due to the drift term is reduced and the basin around the optimal state is mainly sampled due to the noise term, the features of Δ_k for L-FSU ($m = 5, 15$) and CLD only marginally differ. This confirms the earlier findings (see previous paragraphs) that the sampling of this basin is rather independent of curvature-information. Since



(a) *Total chain length l vs iteration index k.*

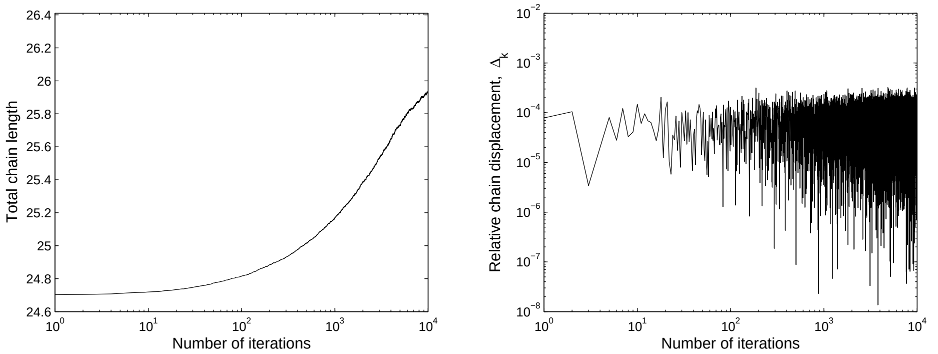
(b) *Relative chain displacement* Δ_k *vs iteration index k.*



(c) *Potential energy difference* $\Delta\Phi$ *vs iteration index k.*

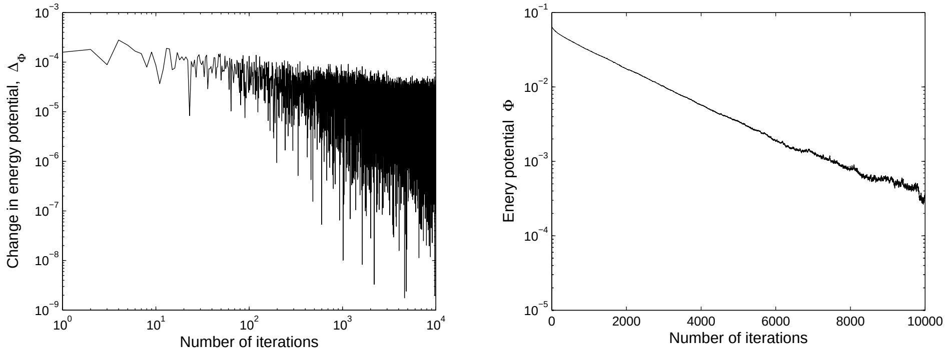
(d) *Potential energy* $\Phi(x_k)$ *vs iteration index k.*

Figure 3.6: *Sampling properties obtained using FSU for the spring potential with* $n = 27$ *and equidistant initial particle distances;* $x_{i,i+1} = 0.95$.



(a) *Total chain length l vs iteration index k.*

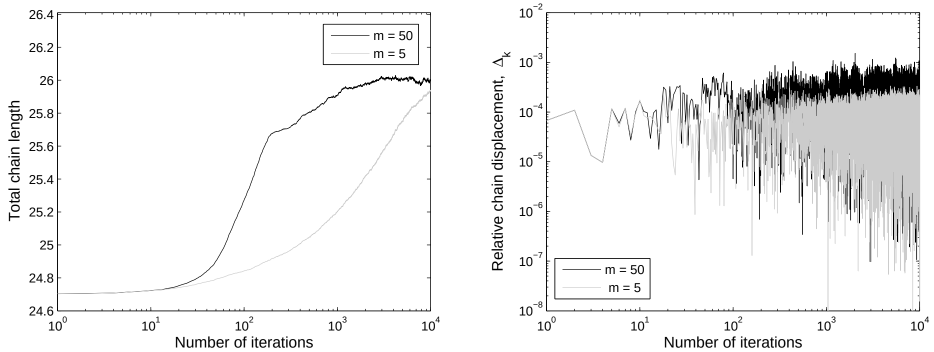
(b) *Relative chain displacement* Δ_k *vs iteration index k.*



(c) *Potential energy difference* $\Delta\Phi$ *vs iteration index k.*

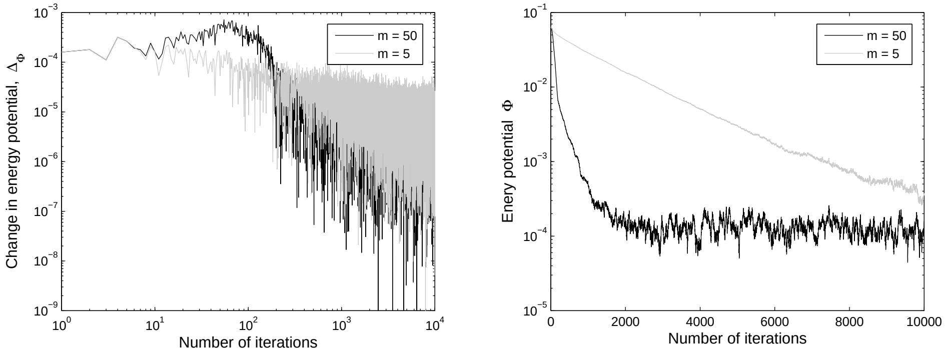
(d) *Potential energy* $\Phi(x_k)$ *vs iteration index k.*

Figure 3.7: *Sampling properties obtained using conventional Langevin dynamics for the spring potential with* $n = 27$ *and equidistant initial particle distances;* $x_{i,i+1} = 0.95$.



(a) *Total chain length l vs iteration index k.*

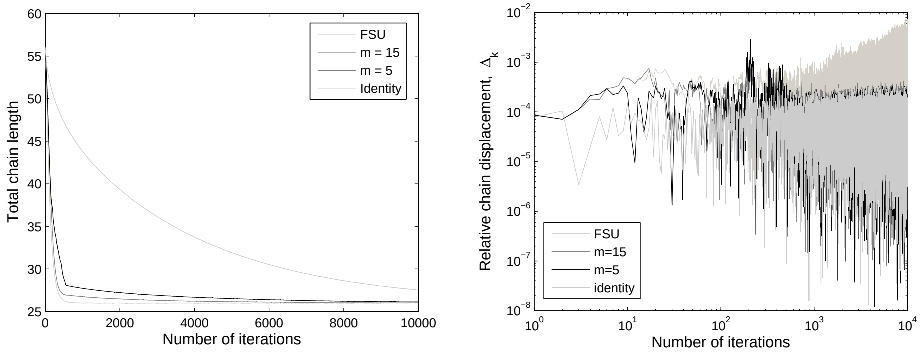
(b) *Relative chain displacement* Δ_k *vs iteration index k.*



(c) *Potential energy difference* $\Delta\Phi$ *vs iteration index k.*

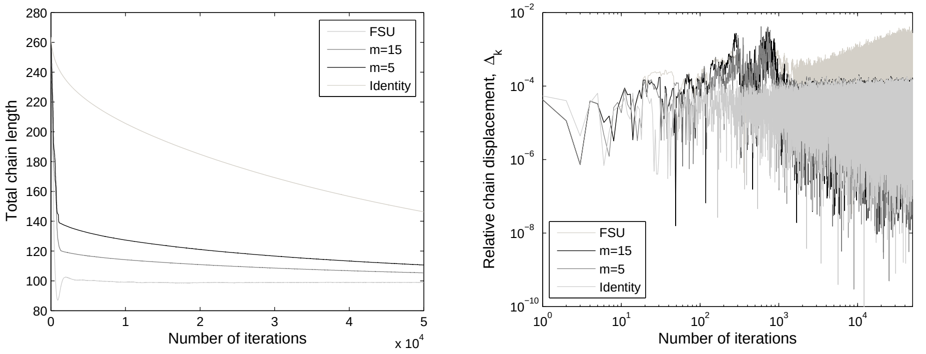
(d) *Potential energy* $\Phi(x_k)$ *vs iteration index k.*

Figure 3.8: *Sampling properties obtained using L-FSU,*$m = 5$ *and* $m = 50$, *for the spring potential with with* $n = 27$ *and equidistant initial particle distances;* $x_{i,i+1} = 0.95$.



(a) *Total chain length l vs iteration index k for n* = 27.

(b) *Relative chain displacement* Δ_k *vs iteration index k for n* = 27.



(c) *Total chain length l vs iteration index k for n* = 100.

(d) *Relative chain displacement* Δ_k *vs iteration index k for n* = 100.

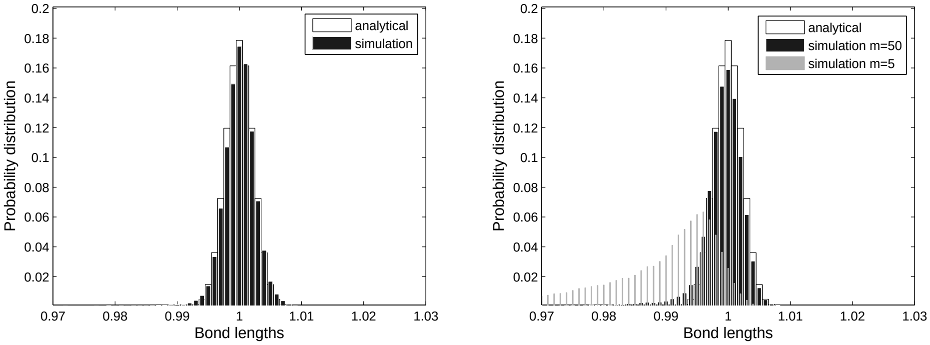
Figure 3.9: *Sampling properties obtained using FSU, L-FSU (m* = 5 *and* 15*) and conventional Langevin dynamics for the spring potential where the particles are initially placed at random interparticle distances between* 0.5 *and* 5.

all spring constants were set to unity, this behavior could be expected (see discussion of the time step). Finally, we compare the sampling performance of FSU, L-FSU and CLD using the same $\mathbf{x}_0$, i.e., random inter-particle distances between 0.5 and 5, for a much larger system, $n = 100$. We find very similar features for the contour length l (Figure 3.9c) and the relative chain displacement (Figure 3.9d) compared to $n = 27$ in Figures 3.9a and 3.9b. However, there are also subtle differences. For FSU, the chain length l mildly overshoots the equilibrium length l_0, directly followed by a correction. This overshoot shows the significance of larger modes. When the length of the chain is close to the equilibrium value, the contributions of larger modes to the particle displacements becomes much smaller but do not completely disappear. Due to the amplification of these modes in FSU, the chain continues to shorten fast. When the chain becomes too short, the forces associated with these modes change sign. Consequently, the chain extends and the chain length convergences to the equilibrium value very fast. For L-FSU ($m = 5$ and $m = 15$) the contribution of the larger modes is damped due to truncation. As a result, the fast decrease of l in the initial stages (first 10^3 steps) due to the larger modes is followed by slower contraction when the contribution due to these modes becomes less significant, and we again observe a distinct kink. The details of Figure 3.9d support this analysis. In particular, the graphs indicate that in the initial stage the relative displacements for L-FSU are considerably larger than for the equivalent simulations with $n = 27$. After this stage, Δ_k for L-FSU and CLD again become rather comparable.

3.3.4. Sampling distribution

We have written $J(\mathbf{x}_k) = J_k$ in equation (3.20) so far. In reality, the iterative update scheme for J introduces memory effects and J contains also previous system states, $J_k = J(\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_k)$. The usual derivation for the Fokker-Planck equation from a stochastic differential equation (sde) like equation (3.6) assumes a Markov process, i.e., no memory effects, but it can be shown that this sde implies the Fokker-Planck equation even when the mobility depends on a finite history of earlier states [72, 73]. In our case, $B_k = J_k J_k^T$ always contains a finite history of earlier states, either due to convergence or truncation. More generally, the number of updates will always be finite. We have previously shown that our general sde indeed gives rise to thermodynamically consistent sampling, i.e. the Boltzmann distribution in equilibrium, for low dimensional systems [46]. We note that the spurious drift term in this general sde gives rise to a predictor-corrector scheme that requires B_k^{-1} (see Appendix B.1). However, this general sde reduces to the discrete equation (3.20) for systems with constant mobility. Although the (inverse) Hessian is a constant for the considered systems, the adaptive mobility B_k is only strictly constant when B_k has converged. It seemed, however, justified to always omit the spurious drift term for the considered quadratic potentials, since either B_k converges fast to a constant value or B_k is very slowly varying along the pathway, in which case the effect of spurious drift is minute and can be neglected (see the results of the previous subsections). Here, we consider the validity of the assumption by analyzing the actual sampling distributions for the systems in the preceding subsection.

In Figure 3.10, we compare the theoretical equilibrium bond-length sampling distribution (TESD) to the simulated sampling distribution using equation (3.20) for 10^4 steps, unless otherwise mentioned. Since all equilibrium bond-lengths are unity, we expect a single peak centered around one. The J_k was obtained by FSU (Figure 3.6) or by L-FSU, with $m = 5$ and $m = 50$ (Figure 3.8). The simulated sampling distribution using FSU in Figure 3.10a fits the TESD very well. Both the spread and the height match perfectly. For L-FSU we expect that an increased number of iterations are required to obtain close to the equilibrium distribution. For $m = 50$ (Figure 3.10b) it is clear that the simulated distribution converges to the TESD, despite a small remaining shoulder due to the limited number of samples. For $m = 5$ this shoulder is more pronounced after 10^4 steps and bond-lengths > 1 are clearly undersampled. Increasing the number of samples will reduce the height of the shoulder in favor of a peak centered around one. We conclude that, also in this case, the TESD will be obtained with increased sampling. Finally, we considered the simulated sampling distribution for $\mathbf{x}_0$ further away from the optimal state using L-FSU ($m = 15$, see Figure 3.9).



(a) *Distribution of the equilibrium bond lengths according to the Boltzmann distribution and calculated distribution after* 10^4 *iterations using FSU.*

(b) *Distribution of the equilibrium bond lengths according to the Boltzmann distribution and calculated distribution after* 10^4 *iterations using L-FSU,* $m = 5$ *and* $m = 50$.

Figure 3.10: *Bond lengths distributions of the spring obtained using FSU and L-FSU (*$m = 5$ *and* 50*) for the spring potential with* $n = 27$ *and equidistant initial particle distances;* $x_{i,i+1} = 0.95$.

Figure 3.11 shows the sampling distribution at different stages: $k = 4000, 6000$ and 10^4. Increasing the number of samples in the simulated sampling distribution leads to a shift of the peak value to one and a decrease of the shoulder on the right. We conclude that also this sampling distribution will converge to the TESD.

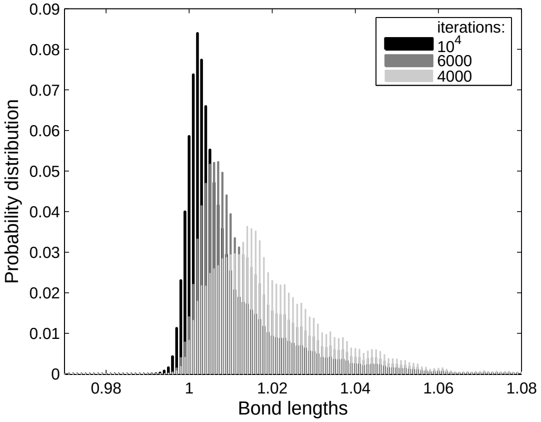


Figure 3.11: *Evolution of the bond length distribution at different stages during a simulation with the spring potential using L-FSU, $m = 15$ and $n = 27$. The particles are initially placed at random interparticle distances between* 0.5 *and* 5.

3.3.5. Discussion

3.3.5a. The inverse Hessian for the unconstrained case

One important issue that is often disregarded in the QN literature is the actual convergence of B_k to the inverse Hessian. As mentioned, the potential Φ may be invariant under particular translations or rotations and H can thus be singular. The mobility B_k may converge to any of the generalized inverses of H, and we should consider $||HB_kH - H||_\mathrm{F}$ to determine convergence properties. Here, we show that it is instructive to consider $||B_kH - I||_\mathrm{F}$ nevertheless, after special taken care of the null-space of H. By definition, $B_kH = I$, or alternatively $H = B_k^{-1}$, if $B_kH\mathbf{x} = I\mathbf{x} = \mathbf{x}$, $\forall \mathbf{x} \in \mathbb{R}^n$. One can easily see that since $B_kH(\mathbf{x} + c\mathbf{1}) = B_kH\mathbf{x}$ for any scalar c, this equality does not have a solution. For the 1-D chain considered in the examples, we can resolve this problem. Since the matrix H is Hermitian positive semi-definite, we can write H as $H = U\Sigma U^T$, with U a unitary matrix and Σ a diagonal matrix containing the singular values σ_i ($i = 1, ..., n$). The columns $\mathbf{u}_i$ ($i = 1, ..., n$) of U form a orthonormal basis for $\mathbb{R}^n$, i.e., $\mathbf{u}_i^T \mathbf{u}_j = 0$ for $i \neq j$ and $= 1$ for $i = j$. Let $\mathbf{u}_n$ be the basis vector associated with the singular value 0 (the null-space). We note that for our system $\mathbf{u}_n = n^{-1}\mathbf{1} = n^{-1}[1 \ldots 1]^T$, due to the invariance of the potential Φ under translation of the whole chain. Writing $\mathbf{x} = \sum_{i=1}^N a_i \mathbf{u}_i$ one easily see that when we add a matrix $B = [\mathbf{u}_n \mathbf{u}_n \ldots \mathbf{u}_n]^T = n^{-1}[\mathbf{1} \ldots \mathbf{1}]^T$ to the left hand side of $B_k H = I$, i.e., $B_k H + B = I$ or equivalent $(B_k H + B)\mathbf{x} = \mathbf{x}\ \forall \mathbf{x} \in \mathbb{R}^n$, this equation can have a solution B_k. We can even make B more general by replacing it by $B = (\sum_i^n \sigma_i')^{-1} \Sigma' \mathbf{1}_M$, where $\mathbf{1}_M$ is a matrix with all unit elements and Σ' is a diagonal matrix. We note that the n values of σ_i' on the diagonal of Σ' can be freely determined, for instance by the requirement that the diagonal elements in $B_k H + B$ are unity. In the examples, we have therefore always considered $||B_k H B_k - I||_\mathrm{F}$ for the unconstrained cases, with σ_i' appropriately chosen. Returning to the original issue of the singularity of H, we see that this procedure essentially allows us to determine an *equivalent* $B_k H$ in which the contribution associated with the null-space of H, i.e., $\mathbf{u}_n$, is removed from each column of $B_k H$. In principle, these contributions can also be obtained from $A = U^T B_k U$, by resetting the entries associated with $\mathbf{u}_n$ to zero in a new matrix $\tilde{A}$. We can then compute $\tilde{B}_k = U \tilde{A} U^T$, followed by an evaluation of $||\tilde{B}_k H - I||_\mathrm{F}$.

3.3.5b. Accelerated sampling and regularization

The idea of reducing or even eliminating critical slowing down by scaling is also present in the Fourier acceleration method for lattice field theory [59]. The application of Fourier acceleration has been extended to Langevin molecular dynamics [60], where acceleration is accomplished by amplifying the slow modes, after transforming the equation of motion to the Fourier domain. Our proposed update does not require any transformation and decoupling of the modes, which is also not always possible in non-trivial models. From the results in the previous sections, we conclude that collective motion, or an effective multiplicity of time steps, is introduced by the new mobility as determined by FSU. Truncation (L-FSU) leads to a slowing down of the collective modes. Although $m = 5$ showed good performance for our particular system, one should take care in determining the optimal history depth m. Since all columns and rows of $\overline{B}(m)$ contain curvature information for $n/2 \leq k \leq n$, a good choice is $m = n/2$. Using this m, the computational costs for L-FSU ($6n^2 + 3n$) are less than FSU ($7n^2$) for $n > 1$. Using (L-)FSU, we obtain an *automated* treatment of different time and length scales in the system. Although the acceleration due to the translational degrees of freedom can be desired in some cases, we showed that it can easily be reduced or even avoided by introducing translational constraints. In general, the null-space related to singular values of H can contain more degrees of freedom than only translation, and it makes sense to constrain them by regularization. We will consider this issue in more detail in the next Chapter.

3.3.5c. Time step

Although the time step Δt in S-QN is at first sight equivalent to the step size α_k in QN methods [46], its role is significantly different. The step size α_k in QN depends on hypersurface information and *varies*. Langevin dynamics, however, is a coarse-grained description for the purely diffusive motion of n particles on a time scale that is large enough to integrate out the rapid modes and replace them by friction and noise. Although this concept can be further exploited for a separation of time scales in the coarse-grained description [74], most conventional Langevin simulations use a single constant time step Δt. In particular, there is no a priori measure based on which one could vary the time step *during* simulation. The time step Δt is lower bounded by the requirement that these rapid modes in the detailed description are always in local thermodynamic equilibrium, and upper bounded by the fastest modes in the diffusive description, in particular by the requirement that these modes are sampled with the correct canonical distribution. In our examples section, we have also used a fixed Δt. The question remains: what is a good value for Δt in S-QN in terms of accurate and efficient sampling? Analysis for a single one dimensional harmonic oscillator, $\Phi = \frac{k}{2}x^2$, with spring constant k provides some insight. An analysis of the time scales involved in the problem using conventional Langevin dynamics gives

$$\frac{d\langle \mathbf{x} \rangle}{dt} = -k\langle \mathbf{x} \rangle \tag{3.25}$$

with $\langle \cdot \rangle$ the ensemble average. We have scaled the time by setting the friction coefficient η to unity. Equation (3.25) shows that when a harmonic degree of freedom is disturbed from equilibrium, it will relax exponentially with a characteristic time k^{-1}. Using S-QN for the same system, i.e., $d\langle \mathbf{x} \rangle/dt = -\langle \mathbf{x} \rangle$, shows that the characteristic time for relaxation that is now the same for any spring constant k, i.e., accelerated for $k < 1$ and slowed down for $k > 1$ with respect to equation (3.25). Further simple analysis of the stability conditions for the discrete time step in equation (3.20) shows that a stable scheme is obtained for $\Delta t < 1$. In comparison, the conventional Langevin equation is stable for $\Delta t < k^{-1}$. The same analysis can be carried out for coupled harmonic oscillators and provides similar results [75]. We note, however, that we have assumed $B = H^{-1}$ and that the analysis is linear. For systems where the higher moments of Φ are significant, a large time step may result in numerical instability and the optimal time step should be determined by numerical testing. Also in the early stages for a quadratic Φ, when the inverse Hessian information stored in B_k is still very sparse, a large time step may give rise to erratic behavior and increased violations of the curvature condition. Since violation of the curvature condition leads to a truncation of updating local Hessian information in B_k, this is very undesired from an algorithmic point of view. In the numerical testing, we found that a large time step (but still $\Delta t < 1$) can indeed initially give rise to features of instability, but that proper behavior is automatically restored at later stages, when the inverse Hessian in increasingly approximated by B_k. We therefore conclude that the time step Δt can in general be much larger than in conventional Langevin dynamics (see also results section). The incorporation of backtracking, similar to QN, could provide a solution to the mentioned problems, but will seriously affect the computationally efficiency and is, if restricted to part of the sampling pathway, rather ad hoc. To resolve the time step issue in the early stages of simulation, it is much simpler to build Hessian information by local sampling, i.e., using equation (3.20) for $\nabla\Phi = 0$ (only noise), and to input the obtained J as J_0 in the FSU method. We believe that the main benefit of our approach originates from introducing a genuine *multiplicity of time steps*, i.e. different scaling for modes in the system depending on information of the inverse Hessian.

3.3.5d. Other energy landscapes

The considered system represents a special case, since H^{-1} is constant. For such systems, the properties of B_k are independent of the sampling part of the energy landscape. For proteins, the potential energy landscape is more complex and commonly referred to as the folding funnel. In particular, it is locally rugged but with an overall funnel shape of the low-energy part, so that most initial conditions are driven toward a 'natural' native state [76]. One may argue that the fast degrees of freedom, such as bond lengths and angles, can be reasonably well described by a harmonic potential energy with different spring constants. However, this is typically not true for all degrees of freedom in a Langevin description. The inverse Hessian will in general significantly change along the sampling pathway, and the approximate B_k has to follow this change accordingly. Consequently, the typical number of steps $k(B)$ for which this memory function can adapt itself to this new situation becomes important. Since the sampling density of the energy landscape is not uniform, the rate $\tau = k(B)/k(H^{-1})$ rather than $k(B)$ itself provides a good estimate, where $k(H^{-1})$ is the typical number

of steps in which H^{-1} changes significantly. We are aware that the definition of these variables is unprecise, but they suffice for our current discussion. We conclude that for $\tau >> 1$, the incorporation of new Hessian information falls behind, and we expect sub-optimal B_k. For the considered harmonic system, $k(B)$ can be seen as the rate of convergence, which depends on the chosen threshold value for the difference norm, see the examples section ($k(B) \sim n^3$). Although $k(H^{-1})$ is infinite in our case, and τ is therefore always smaller than 1, the value of $k(B)$ for a piecewise quadratic Φ may be taken much smaller. Most eigenvalues/eigenvectors of B_k are reasonably approximated in the earlier stages, long before the drop of $||HB_kH - H||_\mathrm{F}$, and one may argue that the actual $k(B) \sim n$. Our results for $n = 27$ and $m = 5$ support this view.
Truncating the history in the L-FSU scheme for constant H^{-1} can have a profound effect, as discarded older information has to be *reconstructed* by the new iterate(s). This is most apparent for small n, where convergence is fast and the majority of information about the inverse Hessian is stored in only a few V_k. Discarding such matrices V_k gives rise to a loss of information, to an extent that the value of m can be detected from periodic steps in the difference norm. The periodicity stems from the periodic removal and restoration of information, since the discarded information is swiftly recovered in the next few steps. For varying H^{-1}, however, memory of unrelated parts in the sampling pathway *should* be removed from B_k, and thus L-FSU for a 'good' choice of m could be more appropriate than FSU and lead to a reduced value of k_B. Since this property only applies for more complex energy landscapes, we will consider this issue in greater detail elsewhere.

3.4. Conclusion

We have constructed a factorized secant update (FSU) for the adaptive compound mobility matrix B in an existing stochastic Quasi-Newton (S-QN) method [46]. The S-QN method is related to the conventional Langevin equation but differs by the fact that curvature information is contained in the mobility via the inverse Hessian, thereby enabling an automated separation of time scales for the different modes in the system. By updating the factorized J instead of $B = JJ^T$ itself, we avoid the cost expensive Cholesky factorization for the noise term in the original scheme and B is always positive definite. The computational costs of FSU are restricted to $7n^2$ multiplications per time step for updating J. In particular, FSU does not require additional evaluations of the potential of interest Φ. For very large n, a limited-memory (L-FSU) update method was derived that allows the user to limit both computational and memory requirements. The approach is based on truncating the memory to m previous updates, similar to the approach in L-BFGS, and requires $12nm + 3n$ multiplications and no matrix storage. The arithmetics of the new L-FSU method is found to be even more optimal than the L-BFGS method. The recursive scheme in L-FSU has the additional advantage to FSU that the initial matrix J_0 is isolated from the rest of the computations, allowing this matrix to be chosen differently in every iteration. We have in detail evaluated the FSU and L-FSU method for a simple but appropriate 1D system of N particles connected by harmonic springs. This system has the advantage of a known analytic Hessian H and involves multiple length and times scales. We analyzed FSU and L-FSU in terms of convergence of B_k to the inverse Hessian, multiscale sampling performance and equilibrium sampling distribution. We found that the B_k determined by FSU converges to a generalized inverse H^- (since H is singular) within $k \sim n^3$ steps. Due to the truncation, the matrix B_k from L-FSU converges to a stationary $\overline{B}(m)$. Analysis of the eigenvalue spectra of B_k suggests that a reasonable good approximate of H^- in FSU is obtained for $k \sim n$. From the analysis of the sampling performance we find that the adaptive mobility indeed gives rise to automated separation of time and length scales. Collective motions, captured by the J_k in both FSU and L-FSU, lead to at least an order of magnitude faster convergence to the lowest energy potential compared to conventional Langevin dynamics. Although for a quadratic Φ good performance was observed for $m = 5$, our analysis suggests that $m = n/2$ is optimal in L-FSU for general Φ. The theoretical (Boltzmann) equilibrium distribution was obtained as a limiting case for both FSU and L-FSU.

CHAPTER 4

Stochastic Quasi-Newton: application to minimal model for proteins

Knowledge of protein folding pathways and inherent structures is of utmost importance for our understanding of biological function, including the rational design of drugs and future treatments against protein misfolds. Computational approaches have now reached the stage where they can assess folding properties and provide data that is complementary to or even inaccessible by experimental imaging techniques. Minimal models of proteins, that enable the simulation of protein folding dynamics by (systematic) coarse-graining, have provided understanding in terms of descriptors for folding, folding kinetics and folded states. Here, we focus on the efficiency of equilibration on the coarse-grained level. In particular, we applied a new regularized stochastic Quasi-Newton (S-QN) method, developed for accelerated configurational space sampling while maintaining thermodynamic consistency, to analyse the folding pathway and inherent structures of a selected protein, where regularization was introduced to improve stability. The adaptive compound mobility matrix B in S-QN, determined by a factorized secant update (FSU), gives rise to an automated scaling of all modes in the protein, in particular an acceleration of protein domain dynamics or principal modes and a slowing down of fast modes or ’soft’ bond constraints, similar to LINCS/SHAKE algorithms, when compared to conventional Langevin dynamics. We used and analyzed a two-step strategy. Owing to the enhanced sampling properties of S-QN and increased barrier crossing at high temperatures (in reduced units), a hierarchy of inherent protein structures is first efficiently determined by applying S-QN for a single initial structure and $T = 1 > T_\theta$, where T_θ is the collapse temperature. Second, S-QN simulations for several initial structures at very low temperature ($T = 0.01 < T_F$, where T_F is the folding temperature), when the protein is known to fold but conventional Langevin dynamics experiences critical slowing down, were applied to determine the protein domain dynamics (collective modes) towards folded states, including the native state. This general treatment is efficient and directly applicable to other coarse-grained proteins.

4.1. Introduction

Efficient sampling of (free) energy landscapes is important in many physical systems, especially when this landscape is very rugged and/or equilibrium states are unknown. In methods that are based on an intrinsic kinetic description, like Molecular Dynamics (MD), the sampling performance is dictated by the smallest time or length scale in the description. As a result, interesting phenomena like the folding of large proteins, in which the scale associated the fastest/smallest (bond vibrations) and the slowest/largest (formation of α or β domains) modes deviate by several orders of magnitude, are inaccessible by standard MD. Slow processes like the cooperative motion of protein domains remain inaccessible even with the increasing computer power, unless some sort of coarse-graining or averaging over the smallest degrees of freedom is carried out. In recent years, approaches based on smoothing, i.e. equilibration on a coarse-grained level followed by fine-grained refinement [79], were developed to address this problem and applied with some success. Several groups have concentrated on determining representative coarse-grained minimal models of proteins. Nevertheless, this approach suffers from a hereditary property, since now the smallest scale on the coarse-grained level determines the sampling performance. Our starting point is conventional coarse-grained Langevin dynamics (CLD), a widely used stochastic model for effective diffusion on a coarse-grained level. We previously showed how to adapt CLD for improved sampling [80]. The general stochastic Quasi-Newton (S-QN) method applies an automated scaling for different length/time scales in the system, by including curvature or correlation information in the space-dependent mobility, while maintaining thermodynamic consistency. Due to the scaling, all modes in the system are updated roughly equally fast, enabling a significantly larger time step (orders of magnitude) compared to CLD. In addition, within this framework CLD reduces to a stochastic form of the well-known steepest descent method. Since quasi-Newton methods are known for their improved ability to locate saddle points compared to steepest descent, also the sampling pathway is positively affected. In [81], we introduced the fundamentals for the efficient determination of $J(\mathbf{x})$ in

$$d\mathbf{x} = [-B(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T \nabla \cdot B(\mathbf{x})]dt + \sqrt{2k_B T}\, J(\mathbf{x}) dW(t), \tag{4.1}$$

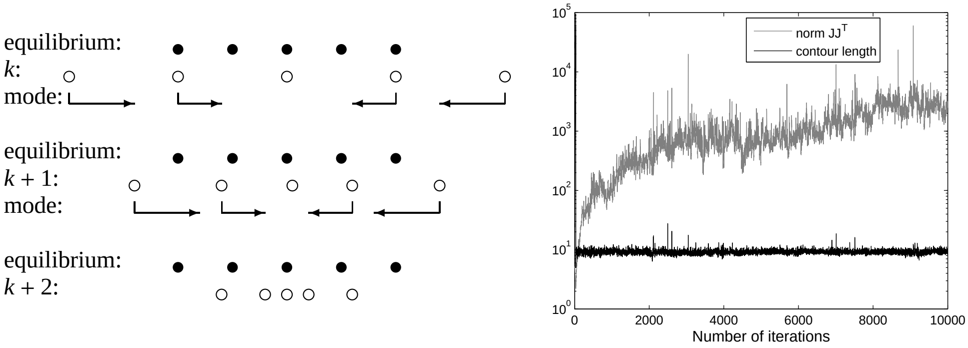
and considered in detail the performance of S-QN for a quadratic energy potentials Φ. In (4.1), the space-dependent mobility $B(\mathbf{x}) = J(\mathbf{x})J(\mathbf{x})^T$ is determined such that it approximates the inverse Hessian. Moreover, the spurious drift term was omitted in Ref. [81] since we only considered quadratic $\Phi(\mathbf{x})$, k_B is the Boltzmann constant,

T the temperature and dW is a Wiener process. Turning to the discrete S-QN equation [81],

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{x}_k - \frac{1}{2}[B(\mathbf{x}'_k)\nabla\Phi(\mathbf{x}'_k) + B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)]\Delta t \\
&\quad + \frac{1}{2}[B(\mathbf{x}'_k)B^{-1}(\mathbf{x}_k) + I]\sqrt{2k_BT}\,J(\mathbf{x}_k)\Delta W_t, \qquad (4.2) \\
\mathbf{x}'_k &= \mathbf{x}_k + \Delta\mathbf{x}^p_k, \qquad (4.3) \\
\Delta\mathbf{x}^p_k &= -B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\Delta t + \sqrt{2k_BT}\,J(\mathbf{x}_k)\Delta W_t. \qquad (4.4)
\end{aligned}$$

the matrix J in $B = JJ^T$ is determined by a rank-two factorized secant update (FSU) scheme. The inverse B^{-1} is determined via an update method in dual space [81]. As discussed before [81], S-QN can be seen as a real-space analogon of existing Fourier acceleration approaches [56–58].

So far, we did not consider a general solution to the problem of ill-conditioning, i.e. when the condition number $\kappa(H)$ of the Hessian matrix $H = H(\mathbf{x})$ becomes very large or when H is singular. For general energy landscapes, this situation is very likely to occur along the sampling path. A straightforward example of such a situation is that Φ is often invariant under a transformation of the whole system, rendering a singular H and a completely flat energy landscape along this eigenvector of H. As B is determined to increasingly approximate the inverse of H, the condition number $\kappa(B)$ will naturally become very large. For this reason, the update may become prone to numerical errors in the update of B and/or $\nabla\Phi$. An additional complication is that the time step in (4.4) is large and that line searches are not included, in contrast to most standard optimization methods [82]. Moreover, for efficiency, our scheme only corrects B in sampled points, based on local Hessian information. When the energy landscape is almost flat along one of the eigenvectors, the large scaling along this direction will give rise to an updated configuration in a distant and possibly uncorrelated part of the energy landscape. As a consequence, the next few updates may suffer from a complex interplay of numerical errors and a sudden discrepancy between B and the actual inverse Hessian. An illustrative sketch of such an interplay is shown in Figure 4.1a, for a Rouse chain with constant but singular Hessian. Since the Hessian is constant, one could understand the origin of this over-acceleration in terms of the constant time step. Upon applying S-QN for the Rouse chain in three dimensions, however, a distinct correlation can be observed between the sudden deviations from the equilibrium chain length and the peaks in $||B||_F$ (Figure 4.1b) that start to appear for rather large $||B||_F$. Close examination indicated that these chain length deviations



(a) *Over acceleration of the slow mode.*

(b) $||JJ^T||_\mathrm{F}$ *and contour length, where the peaks in the contour length coincide with the peaks in* $||JJ^T||_\mathrm{F}$*; equilibrium chain length is 9.*

Figure 4.1: *Reasons regularization needed: over amplification of the slow modes and erroneous approximation of the eigenvectors.*

are due to numerical errors in updating B. In particular, Φ is invariant under translation of the whole chain, which is the eigenvector $\mathbf{1} = [1...1]^T$ of H with eigenvalue zero. In terms of the (generalized) inverse of H, this eigenvector is associated with the largest eigenvalue (or scaling). After determining the eigenvector associated with the largest eigenvalue for each B, the sudden increase in chain length and norm of B was found to coincide with a slight deviation of this eigenvector from $\mathbf{1}$. Due to large scaling along this direction, deviations that exist for some particles along the chain will be substantially amplified and result in chain extension or compression. Nevertheless, the equilibrium chain length was recovered quickly after chain extension or compression. A way to resolve both discussed issues above simultaneously is to constrain the step size or, equivalent, condition the matrix B by regularization.

Our main goal in this Chapter is to apply the general S-QN method for a system of obvious physical relevance. Minimal models for proteins have the advantage that systematic coarse-graining procedures gave rise to rather simple but accurate energy expressions and that folding pathways and folded states for a number of proteins were determined both experimentally and by simulation [83–91]. Fast and accurate determination of folded states and primary modes along the folding pathway is not only interesting from a scientific point of view, this system also represents a good benchmark for S-QN because of the complex energy landscape topography and the fact that conventional Langevin dynamics is often employed in current studies [92, 93]. We choose one of these proteins, containing 22 amino acids, as a reference system.

First, we introduce the new regularization parameter ϵ and show that the regularized B^{-1} converges to $H + \epsilon I$, thereby resolving both the problem of conditioning B and the singularity of H. Since regularization gives rise to only a slight modification of the FSU scheme, the efficiency of the original scheme ($8n^2$) is retained. Although we will not consider limited-memory implementation here, the approach can easily be extended to L-FSU [81]. We analyze the effect of regularization and apply the new scheme to the minimal model of the chosen protein in a stepwise manner, by including an increasing number of terms of the total coarse-grained energy expression. We conclude with a detailed analysis of the collective dynamics and sampling behavior for the considered coarse-grained protein.

4.2. Theory

Our factorized secant update (FSU) scheme for updating the factorized term is given by

$$J_{k+1} = J_k + \frac{\alpha_k \mathbf{s}_k \mathbf{y}_k^T J_k - \alpha_k^2 J_k J_k^T \mathbf{y}_k \mathbf{y}_k^T J_k}{\mathbf{y}_k^T \mathbf{s}_k}, \tag{4.5}$$

with

$$\alpha_k^2 = \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{y}_k^T J_k J_k^T \mathbf{y}_k}, \tag{4.6}$$

where $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$, $\mathbf{y}_k = \nabla\Phi(\mathbf{x}_{k+1}) - \nabla\Phi(\mathbf{x}_k)$ and $B_{k+1} = J_{k+1} J_{k+1}^T$. We have previously shown that (4.5) is equivalent to the standard Davidon-Fletcher-Powell (DFP) update formula [81],

$$B_{k+1} = B_k - \frac{B_k \mathbf{y}_k \mathbf{y}_k^T B_k}{\mathbf{y}_k^T B_k \mathbf{y}_k} + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}. \tag{4.7}$$

For a regular Hessian H, the DFP method is known to possess three important properties [66]: a) positive-definiteness of the update matrix is assured provided that B_0 is chosen to be positive definite, b) for a positive-definite quadratic objective function

$\Phi(\mathbf{x})$, with $\mathbf{x}$ an n-dimensional vector, the algorithm converges to the solution in at most n steps, and c) for a positive-definite quadratic objective function $\Phi(\mathbf{x})$, and if the convergence to the solution requires the full n steps, $B_n = H^{-1}$ where the matrix H is the exact inverse Hessian.

In particular property a) is a prerequisite for our method, due to the presence of the noise term containing J with $JJ^T = B$. Convergence properties b) and c) are very sensitive to the line search accuracy, i.e. the determination of the step size α_k such that Φ is minimal along the search direction or, alternatively, that all search directions $\mathbf{s}_k = \alpha_k B_k \nabla\Phi$ are orthogonal. Since our $\mathbf{s}_k$ contains *two* terms, scaling as α_k and $\sqrt{\alpha_k}$ respectively, and since J is not always available as a matrix, the step size in our method, the physically relevant time step Δt [81], was chosen to be a *constant*. In contrast to QN methods, where the condition for the step size is either an exact line search, a reduction of $||\Phi||$ or the equivalence to Newton's method ($\alpha_k = 1$) [94], the time step in S-QN is also physically constrained. Our time step should be such that the fastest mode in the system is well represented for $M(\mathbf{x}) = B_0 = I$, the often considered initial value for B. For quadratic positive-definite Φ, or for Φ that are locally well approximated by a quadratic function, the fastest mode is associated with the largest eigenvalue $\lambda_{\max}(H)(>> 1)$ of the Hessian H and the time step Δt is thus upper bounded by $\lambda_{\max}(H)\Delta t < 1$ as a rule of thumb. Only when the mobility is a good approximate of the inverse Hessian, the upper bound increases to $\Delta t < 1$ and we obtain improved sampling efficiency.

For a quadratic Φ, we have previously shown that B_k determined by FSU with a constant time step does converge to the generalized inverse Hessian H^- in $O(n^3)$ steps, instead of the theoretical quadratic convergence ($O(n)$) in property c) [81]. However, one should consider that this H was *singular* and that the convergence of the non-extremal eigenvalues of B was actually much faster than $O(n^3)$. The invariance of Φ under certain fully cooperative displacements, such as a translation or a rotation with respect to the center of mass, is a common property and gives rise to a singular H. As a result of $\lambda_{\min}(H) = 0$ and the rank-two update formula, the extremal eigenvalues of B continued to increase/decrease with increasing k and the condition number of B, $\kappa(B) = \lambda_{\max}(B)/\lambda_{\min}(B)$, became arbitrarily large. In general, a *low* condition number $\kappa(B)$ is desirable because of possible round-off errors in the update scheme [95]. As $\lambda_{\max}(B)$ becomes very large, any error in the associated eigenvector is seriously amplified. Moreover, since this displacement is very large, the condition number $\kappa(B_k)$ may be much larger than the condition number of the inverse Hessian $G(\mathbf{x}_{k+1})$ in the updated position for general Φ. It is unclear if the DFP scheme can recover [96].

4.2.1. Existing approaches for conditioning B

Solutions to both problems, the exact line searches and conditioning, have been considered in the QN literature. The attempts to avoid ill-conditioning mainly focus on quadratic and convex objective functions and also the analysis was carried out under conditions which are typical for quasi-Newton methods. Many of the algorithms without line searches are rank-one update formulas that do not always satisfy the required property a), i.e. a positive-definite B, and may even give rise to singular updates [97]. The most successful rank-two approach without exact line searches relies on the convex class of update formulas given by [96]

$$B_{k+1}(\phi) = (1-\phi)B_{k+1}^{\mathrm{DFP}} + \phi B_{k+1}^{\mathrm{dual}} = B_{k+1}^{\mathrm{DFP}} + \phi \mathbf{v}_k \mathbf{v}_k^T \tag{4.8}$$

where B_{k+1}^{DFP} is the DFP update given by (4.7) and

$$\begin{aligned} B_{k+1}^{\mathrm{dual}} &= (I - \frac{\mathbf{s}_k \mathbf{y}_k^T}{\mathbf{s}_k^T \mathbf{y}_k}) B_k (I - \frac{\mathbf{y}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k}) + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} \\ \mathbf{v}_k &= (\mathbf{y}_k^T B_k \mathbf{y}_k)^{1/2} \{ \frac{\mathbf{s}_k}{\mathbf{s}_k^T \mathbf{y}_k} - \frac{B_k \mathbf{y}_k}{\mathbf{y}_k^T B_k \mathbf{y}_k} \} . \end{aligned} \tag{4.9}$$

The update B_{k+1}^{dual} is dual to DFP in the sense that it is derived from (4.7) by inversion via the Sherman-Morrison theorem and subsequently interchanging $\mathbf{y}_k$ and $\mathbf{s}_k$. The free parameter ϕ is determined based on update properties and should be chosen $0 \le \phi \le 1$ to retain convergence property c) [96, 98]. This class of update formulas was found to be more robust to the update becoming singular or unbounded than the individual methods for $\phi \in \{0, 1\}$ in the case of inexact line searches. Later contributions have adapted (4.8) for optimal conditioning of B (see [99] for details). In *sizing*, the objective function Φ is multiplied with a scalar parameter, based on the recognition that the update scheme is invariant under this scaling. The second method is *switching*, where this switch is between different methods of the convex class (4.8).

Issues of instability and sensitivity to (numerical) noise also occur in *ill-posed* problems in mathematics. Common strategies to deal with these issues include reducing the dimensions of the system, e.g. using different refinements: rigid body, torsion and angle, and removing or adapting the small eigenvalues of the system. A number of established methods, including truncated singular value decomposition, truncated total least square, generalised singular value decomposition, truncated rank-revealing decomposition and Tikhonov regularization, exist for this purpose. Physically-inspired regularization uses prior knowledge of the system to convert an ill-posed problem (or singular problems) into regular problems. A way to do this is to define additional constraints, for instance by prohibiting atoms from getting too close to each other. In our Rouse chain [81], constraints based on the null space of H could be applied, by subtracting translations and rotations in the displacement of the total chain. Translations are treated by keeping the chain center of mass fixed, and subtracting this part of the displacement in (4.1). Subtracting chain rotation can be done by finding a rotation matrix R, such that the root mean square deviation (RMSD), defined as

$$\text{RMSD} = \min_{R} \sum_{i}^{N} ||R\tilde{\mathbf{r}}_i^k - \mathbf{r}_i^k|| \tag{4.10}$$

is minimal, where $\tilde{\mathbf{r}}_i$ is the position of particle i at iteration k after adding the Langevin displacement and after translation of the chain back to its original center of mass. The new position of particle i becomes $\mathbf{r}_i^{k+1} = R\tilde{\mathbf{r}}_i^k$. Obviously, determining R by straightforward minimization is inefficient, but quaternion algebra provides an efficient alternative [100].

4.2.2. RFSU: a regularized FSU method

Since proper scaling of different modes is key to our approach, we consider the ill-conditioned or singular H and hence the arbitrary large condition number $\kappa(B)$ the most important issue. We further rely on the observation that the convergence for the non-extreme eigenvalues of B for the quadratic Φ in Ref. [81] was relatively fast and also the free parameter ϕ in the convex class of Broyden methods is determined based on conditioning properties. Moreover, by conditioning (4.7) we avoid the nontrivial task of deriving a factorized update algorithm for (4.8) and determining a good value of ϕ for general objective functions Φ (remember that $\phi = 1$ is the DFP method). We could in principle use *sizing* for conditioning B determined by (4.7), but this may not help since H itself is singular. Instead, we regularize the DFP scheme using an approach that is conceptually very similar to Tikhonov regularization.

The key idea is very simple: we adapt the update scheme such that $\tilde{B}_k^{-1}$ converges to $\tilde{H} = H + \epsilon I$, with ϵ a (small) regularization parameter. It is easy to show that the eigenvalues $\{\lambda_1(H), ..., \lambda_n(H)\}$ shift to $\{\lambda_1(\tilde{H}), ..., \lambda_n(\tilde{H})\} = \{\lambda_1(H) + \epsilon, ..., \lambda_n(H) + \epsilon\}$. For $\epsilon > 0$, all eigenvalues of $\tilde{H}$ are non-zero and the condition number becomes bounded, i.e. $\kappa(\tilde{H}) = \lambda_{\max}(H)/\epsilon + 1$, since $\lambda_{\max}(H)$ is bounded. Similar regularization was previously introduced [101] to deal with negative curvatures, but relied on compute-intensive explicit determination of eigenvalues of B via LDL^T factorization. Although the key idea is simple, we have to adapt our update formula for B in dual space. Let B_{k+1} of (4.7) be the update. The inverse G_{k+1} of B_{k+1} in the dual space is explicitly given by (4.9) after interchanging $\mathbf{y}_k$ and $\mathbf{s}_k$:

$$G_{k+1} = (I - \frac{\mathbf{y}_k\mathbf{s}_k^T}{\mathbf{y}_k^T\mathbf{s}_k})G_k(I - \frac{\mathbf{s}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\mathbf{s}_k}) + \frac{\mathbf{y}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\mathbf{s}_k} \tag{4.11}$$

It is easy to see that the secant condition in the dual space, $G_{k+1}\mathbf{s}_k = \mathbf{y}_k$, is satisfied. Following an idea introduced by Powell for treating the ill-conditioned case in constrained optimization [102, 103], we note that any matrix $\tilde{G}_{k+1}$ obtained by (4.11) after replacing $\mathbf{y}_k$ by $\tilde{\mathbf{y}}_k$ satisfies the secant condition $\tilde{G}_{k+1}\mathbf{s}_k = \tilde{\mathbf{y}}_k$. Let again $\tilde{G}_{k+1}$ be the regularized matrix $G_{k+1} + \epsilon I$. Since G_k converges to H, $\tilde{G}_k$ converges to $\tilde{H} = H + \epsilon I$. We can now determine $\tilde{\mathbf{y}}_k$ from

$$\tilde{\mathbf{y}}_k = \tilde{G}_{k+1}\mathbf{s}_k = G_{k+1}\mathbf{s}_k + \epsilon\mathbf{s}_k = \mathbf{y}_k + \epsilon\mathbf{s}_k. \tag{4.12}$$

After inversion, we obtain the update scheme

$$\tilde{B}_{k+1} = \tilde{B}_k - \frac{\tilde{B}_k\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k^T\tilde{B}_k}{\tilde{\mathbf{y}}_k^T\tilde{B}_k\tilde{\mathbf{y}}_k} + \frac{\mathbf{s}_k\mathbf{s}_k^T}{\tilde{\mathbf{y}}_k^T\mathbf{s}_k} \tag{4.13}$$

with $\tilde{\mathbf{y}}_k = \mathbf{y}_k + \epsilon\mathbf{s}_k$ that satisfies the secant condition $\tilde{B}_{k+1}\tilde{\mathbf{y}}_k = \mathbf{s}_k$. Since $\tilde{\mathbf{y}}_k^T\mathbf{s}_k = \mathbf{y}_k^T\mathbf{s}_k + \epsilon\mathbf{s}_k^T\mathbf{s}_k$ and $\mathbf{s}_k^T\mathbf{s}_k$ is always positive, a possible cause of numerical errors in the original scheme, i.e. $\mathbf{y}_k^T\mathbf{s}_k$ is positive but very small, is avoided. We only left with the proof that $\tilde{B}_k$ converges to $[H+\epsilon I]^{-1}$. We follow the proof given in Ref. [96] (Section 6) and assume that the objective function is quadratic positive-definite. Setting $K = [H + \epsilon I]^{1/2}\tilde{B}[H + \epsilon I]^{1/2}$ and denoting K_{k+1} and K_k by K^* and K respectively, we can rearrange (4.13) into

$$K^* = K - \frac{K\mathbf{z}\mathbf{z}^T K}{\mathbf{z}^T K\mathbf{z}} + \frac{\mathbf{z}\mathbf{z}^T}{\mathbf{z}^T\mathbf{z}} \tag{4.14}$$

where $\mathbf{z} = [H + \epsilon I]^{1/2}\mathbf{s}_k$ and $\mathbf{y}_k = H\mathbf{s}_k$. In particular, $\mathbf{z}$ is an eigenvector of the first two terms in (4.14) with $\lambda = 0$ and the third term sets this eigenvalue to $\lambda = 1$ while leaving the others unchanged. Since relation (4.14) is exactly the same as in Ref. [96], the convergence of $\tilde{B}$ to $[H+\epsilon I]^{-1}$ is proven. The matrix $\tilde{H}$ is now regular, so $[H + \epsilon I]^{-1} = H^{-1} - \epsilon H^{-1}H^{-1} + O(\epsilon^2)$ exists and is well defined. Finally, we shortly consider the condition number of $\tilde{B}_{k+1}$. Let $\mathbf{v}$ be a eigenvector of G_{k+1} with eigenvalue λ. It is easy to see that $\mathbf{v}$ is also an eigenvector of B_{k+1} with eigenvalue $1/\lambda$. Moreover, $\mathbf{v}$ is an eigenvector of $G_{k+1} + \epsilon I = \tilde{G}_{k+1}$ with eigenvalue $\lambda + \epsilon$ and of $\tilde{B}_{k+1}$ with eigenvalue $1/(\lambda + \epsilon)$. Hence, the condition number $\kappa(\tilde{B}_{k+1}) = (\lambda_{\max}(G_{k+1}) + \epsilon)/\epsilon$ with ϵ a freedom of choice. In other words, the condition number and the maximum displacements have become bounded when $\lambda_{\max}(G_{k+1})$ is finite. A factorized $\tilde{J}_{k+1}$ such that $\tilde{J}_{k+1}\tilde{J}_{k+1}^T = \tilde{B}_{k+1}$ is found by (4.5) after replacing $\mathbf{y}_k$ by $\tilde{\mathbf{y}}_k$.

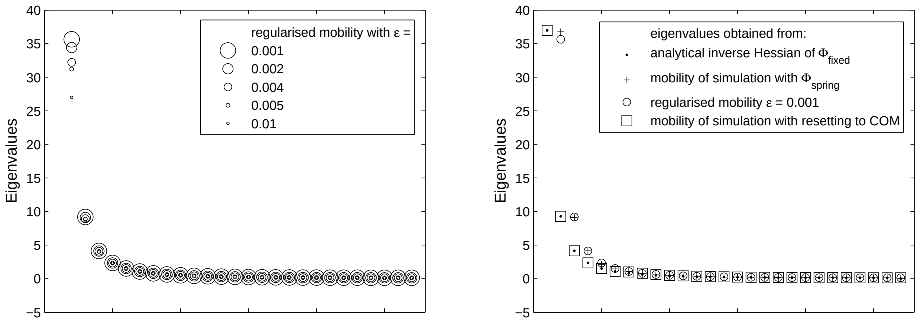
4.3. Results and discussion

4.3.1. The choice of the regularization parameter

A good choice of ε depends on properties of the system, in particular on the time step Δt and on the smallest eigenvalues of the Hessian H. Since H can easily become singular, a useful rule of thumb is that the step size on the energy landscape is maximized by RFSU, by a factor $\equiv \Delta t/\varepsilon$. Small ε only slow down the largest collective displacements, while for very large ε, the regularization term in B starts to dominate and RFSU will resemble conventional Langevin dynamics with constant friction $1/\varepsilon$. Here, we start by analyzing RFSU for the 1-D chain of our previous paper [81] with a harmonic potential

$$\Phi_{\text{spring}} = \sum_{i=1}^{n-1} (x_{i,i+1} - 1)^2; \tag{4.15}$$

and $x_{i,i+1} = |x_i - x_{i+1}|_2$ the distance between particle i and $i+1$. Since the Hessian for this potential is constant, we can analyze the converged eigenvalues of B for different ε. We note that these eigenvalues should converge to $(\lambda(H)+\varepsilon)^{-1}$ (see theory section). Figure 4.2a shows that choosing different ε indeed does not have a noticeable effect on the smallest eigenvalues. As expected, the effect is most significant for the larger eigenvalues ($\equiv$ small $\lambda(H)$) and the largest eigenvalue always converged to $1/\varepsilon$ (remember that $\lambda^{\min}(H) = 0$). In Figure 4.2b we compare the eigenvalue spectra of B_ε ($\circ$) using RFSU for $\varepsilon = 0.001$ and B_Φ (+) using FSU for $n = 27$. We previously showed [81] that B_Φ converges to a generalized inverse of H. After 10000 steps, the largest eigenvalue of B_ε has converged to $1/\varepsilon = 10^3$ while the largest eigenvalue of B_Φ is still increasing [81] (both exceed the vertical axes limit). In particular, B obtained by RFSU converges to a unique $(H + \varepsilon I)^{-1}$, the rate of which depends on the value of ε. Additional symbols in Figure 4.2b denote the eigenvalues spectra obtained for (4.15) by analytic inversion and for two types of physically-inspired regularizations. First (□), we explicitly constrained FSU for chain translation by resetting the center of mass to the original position at each time step. Second (·), we calculated the analytic eigenvalue spectrum for (4.15) with an additional penalty function for the center of mass, similar to earlier work [81]. It is clear that these two types of regularizations have a very similar effect, since the eigenvalue spectra overlap. However, both spectra differ from RFSU. Removing or penalizing the slowest mode in the displacements apparently slows down *all* modes compared to the non-regularized case. In particular, the eigenvalue spectrum seems shifted (by one) when one compares the spectra for the non-regularized (+) and the physically-inspired regularized results (□ and ·). In contrast, the spectrum for RFSU and non-regularized FSU share many features. In RFSU the acceleration of the slowest modes is maximized and the condition number $\kappa(B)$ is therefore constrained. We also review the effect of



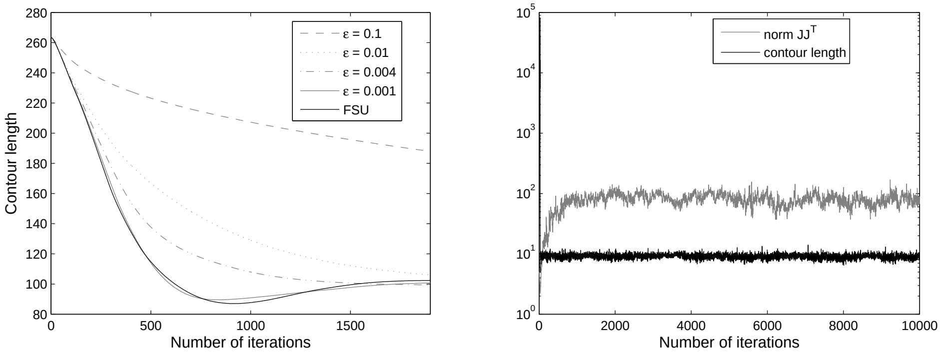
(a) *Eigenvalues of mobility constructed with regularized FSU for different values ε.*

(b) *Eigenvalues of the mobility constructed with different regularization methods.*

Figure 4.2: *Comparison of the eigenvalues of the analytic inverse Hessian and the mobility as constructed by regularized FSU and other regularization methods with* $n = 27$.

regularization on the ill-conditioning and over-acceleration issues discussed in the introduction (see Figure 4.1). We consider a 1-D chain (4.15) for $n = 100$ for over-acceleration [81] and the same 3-D Rouse chain as in Figure 4.1 for ill-conditioning. Figure 4.3a shows the evolution of the chain length for different values of ε, including $\varepsilon = 0$ of FSU. Over-acceleration persists in RFSU for small values of ε, as the chain length falls substantially below the equilibrium length $l_{eq} = 99$ before converging.

For the largest considered value, $\varepsilon = 0.1$, convergence to l_{eq} is considerably slowed down and approaches the one obtained for conventional Langevin dynamics [81], since the diagonal regularization matrix starts to dominate. Here, the contribution of collective modes is severely damped by maximizing the largest eigenvalues of B to $1/\varepsilon = 10$, resulting in a much slower cooperative chain contraction than in FSU. Proper behavior is obtained somewhere between these two extremes, in particular for $0.025 < \varepsilon < 0.01$, where the chain length convergence is even better than that of FSU. We found that $\varepsilon = 0.004$ is optimal. Application of RFSU also resolves the problem of an ill-conditioned B. Upon comparing the matrix norm of B in Figure 4.1b for FSU and Figure 4.3b using RFSU with $\varepsilon = 0.01$, one can observe that the sudden increases in the chain length are completely absent after regularization. Clearly, the effect of amplification of numerical errors in B is no longer present or considerably reduced.



(a) *Effect of the regularization on the convergence towards the equilibrium contour length of 3-D Rouse chain with equilibrium length 99.*

(b) *Effect of the regularization on* $\|JJ^T\|_{\mathrm{F}}$ *and the contour lengths during the simulation of a 3-D Rouse chain with equilibrium length 9.*

Figure 4.3: *Examples of the regularization effect on over acceleration and on amplification of numerical errors.*

4.3.2. Minimal model for a protein

The development of coarse-grained or minimal models for proteins has always been a very active area of research. Besides the intellectual challenge, a solution to the protein folding problem will have important implications for the understanding and design of protein function and possible causes of diseases due to protein misfolds. Experiments show that small, single-domain proteins reach their native states on the time scale in the order of 10-1000 milliseconds, thereby setting a lower bound for the representation of the smallest modes. The effect of smoothing the energy landscape by coarse-graining was previously discussed (we refer to existing reviews [104] for details) and coarse-grained models are believed to provide valuable information of folding, assembly and function(s) of biomolecules.

Since the earliest efforts of Flory for heterogeneous polymers [105], several groups have focussed on a coarse-grained off-lattice C_α representation of proteins. Such a representation disregards side chains effects and introduces a three letter particle code for each peptide depending on their nature: *hydrophobic*(B), *polar or hydrophilic*(L) and *neutral*(N). A general expression for the conformational energy potential Φ_{p} for a string of particles is then given by

$$\Phi_{\mathrm{p}} = \Phi_{\mathrm{bond}} + \Phi_{\mathrm{bending}} + \Phi_{\mathrm{dihedral}} + \Phi_{\mathrm{non}}, \tag{4.16}$$

where the terms are the potential for bond length variation, which can be omitted if the bond length is fixed [109], bond angles, dihedral angles and nonbonded interactions. Using this model, several sequence-dependent properties like folding trajectories, metastable fold states and life-times, the effect of mutations and thermodynamic properties like folding and collapse temperatures, in order to discriminate between fast and slow folders, have been considered and compared to existing experimental observations [91].

A realistic kinetic description is vital for these studies and most of them rely on Langevin dynamics, either in the over- or under-damped limit. Our S-QN model considers Langevin dynamics in the overdamped case as a starting point but does not follow a realistic pathway due to the incorporation of curvature information. This is a property shared among many other methods, for instance Monte Carlo (MC) and methods that employ a biasing potential [106], aimed at improved sampling. However, S-QN does allow for an automated determination of collective modes and is much more efficient than Langevin dynamics in finding transition and equilibrium states in a thermodynamically consistent way, as we will show in this section for a particular coarse-grained protein. Due to these unique properties, we believe that this model represents a valuable contribution to the coarse-grained protein modeling toolkit.

4.3.2a. Terms in the energy potential

The bond potential is given by

$$\Phi_{\text{bonded}} = \sum_{i=1}^{N-1} k_{ij}^{b}(d_{i,i+1} - d_0)^2, \tag{4.17}$$

with N the number of particles, $j = i + 1$, $d_{i,i+1} = \|\mathbf{r}_i - \mathbf{r}_{i+1}\|_2$ is the distance between particle i and $i + 1$, d_0 is the reference bond length and k_{ij}^{b} is the force (or spring) constant. The angle or bending potential is given by

$$\Phi_{\text{bending}} = \sum_{i=1}^{N-2} k_{ijk}^{a}(\theta_{ijk} - \theta_0)^2, \tag{4.18}$$

with $j = i + 1$, $k = i + 2$, θ_{ijk} the bond angle defined by the bond vectors $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ and $\mathbf{r}_{kj} = \mathbf{r}_k - \mathbf{r}_j$, θ_0 is the reference angle and k_{ijk}^{a} is the force constant. The torsion potential considered in the next paragraph is very general,

$$\Phi_{\text{torsion}} = \sum_{i=1}^{N-3} k_{ijkl}^{t}(1 + \cos(n_{ijkl}\phi_{ijkl} - \phi_0)), \tag{4.19}$$

with $j = i + 1$, $k = i + 2$, $l = i + 3$, ϕ_{ijkl} the torsion angle between the normal of the plane through particles i, j, k and the normal of the plane through particles j, k, l. Again k_{ijkl}^{t} is the force constant and ϕ_0 is the reference angle and n_{ijkl} is the multiplicity. In the simulations of the specific protein, we apply a somewhat different potential [91]

$$\Phi_{\text{dihedral}} = \sum_{i=1}^{N-3} \left[A_i(1 + \cos\phi_i) + B_i(1 + \cos 3\phi_i)\right], \tag{4.20}$$

The non-bonded potential is of Lennard-Jones type and given by

$$\Phi_{\text{non}} = \sum_{i=1}^{N-3} \sum_{j=i+1}^{N} 4\left[C_{ij}\left(\frac{d_0}{d_{i,j}}\right)^{12} - D_{ij}\left(\frac{d_0}{d_{i,j}}\right)^{6}\right], \tag{4.21}$$

4.3.2b. Preliminary calculation of sampling distributions

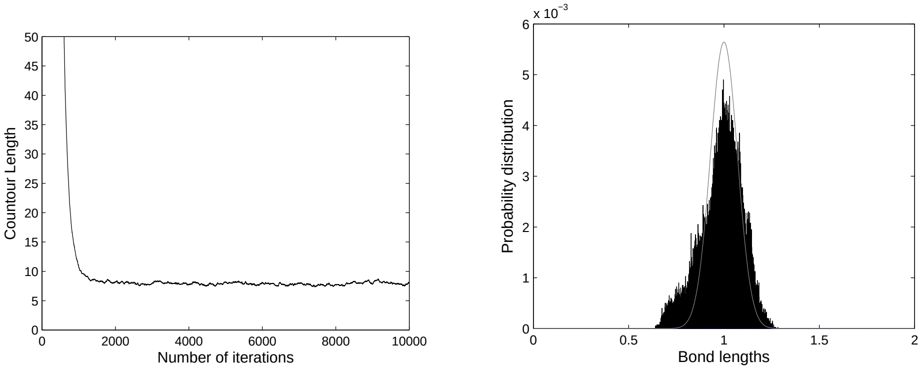
First, we consider the thermodynamically consistency of the sampling obtained by RFSU, which should be the Boltzmann distribution in equilibrium, for the case where only intra-molecular interactions contribute. Sampling distributions for the bond length, angle and torsion, for the torsion potential (4.19), are considered separately using a simulation of 10000 steps in total. We apply RFSU for a string of $N = 9$ equivalent particles, $n = 3 \times N = 27$, and use $k_B T = 10$ and $\Delta t = 10^{-2}$. The force constants are set to $k^b_{ij} = k^a_{ijk} = k^a_{ijkl} = 10^3$ and the reference variables $d_0 = 1$ and $\theta_0 = \phi_0 = \frac{\pi}{2}$. We varied ε and the initial configurations. The presented results are for one initial configuration and rather high $\varepsilon = 0.1$, but representative for all other cases.

The chain length during the simulation, Figure 4.4a, converges to an equilibrium $l_{eq} = (N-1)d_0 = 8$ after roughly 1000 iterations. The analytic Boltzmann distribution was calculated using $\mathcal{N}\exp(-\Phi/k_B T)$, where Φ is the considered potential term. Starting with the bond length, we see in Figure 4.4b that the analytic distribution is reproduced well by the simulated sampling distribution, with mean $d_0 = 1$. From Figures 4.4c and 4.4d we conclude the same for the simulated distribution of the bending angles and dihedrals, respectively. Apparently, S-QN is effective in sampling the equilibrium values for the bond length, angles and dihedrals, in spite of the rather large ε.

We note that the update is governed by the noise term for the chosen value of $k_B T = 10$ (the noise/drift ratio >> 1). Although slow modes are seriously damped due to regularization, relative to the non-regularized case, the collective contributions of noise and drift enable this effective sampling. As a control, we also performed a simulation using conventional Langevin equation, with a constant mobility $M = 1$ (all other parameters are the same). After 10000 steps, the sampling distribution is far from equilibrium and preferable bond length, bending and torsion angles were not detected.

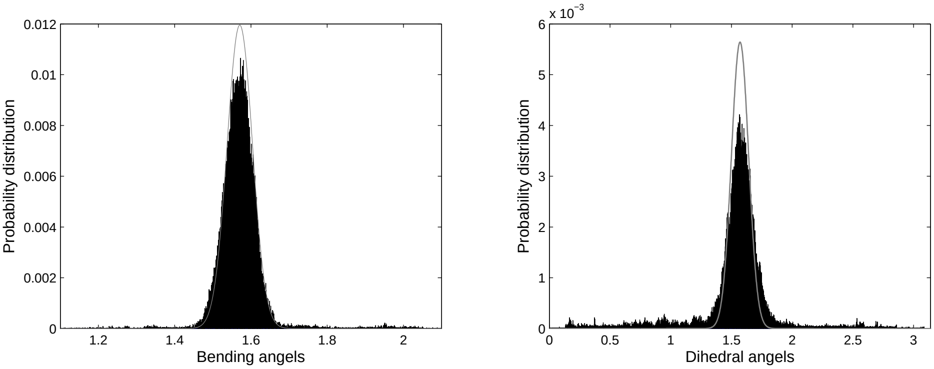
4.3.2c. Application to the model protein

Finally, we apply RFSU to an earlier considered coarse-grained protein G [91], represented by the string $LB_9(NL)_2NBLB_3LB$, where each letter denotes a hydrophobic (B), hydrophilic (L) or neutral (N) coarse-grained peptide. The simulations in Ref. [91] were performed using Langevin dynamics in the under-damped limit for different friction coefficients. Since the value of $\sigma = T_\theta - T_F/T_\theta$ for G is rather low ($\sigma = 0.20$), with $T_\theta = 0.78$ the collapse transition and $T_F = 0.62$ the folding transition temperature (in reduced units of ε_h/k_B, with ε_h a scaling factor in the non-bonded



(a) *Total chain length vs iteration index k*

(b) *Distribution of the bond lengths according to the Boltzmann distribution and the calculated distribution using regularized FSU.*



(c) *Distribution of the angles according to the Boltzmann distribution and the calculated distribution using regularized FSU.*

(d) *Distribution of the dihedrals according to the Boltzmann distribution and the calculated distribution using regularized FSU.*

Figure 4.4: *Sampling properties of the contour length, bond length, bending angles and dihedrals using regularized FSU with $N = 9$ (i.e. n=27), $\varepsilon = 0.01$, $k_B T = 1$ and $\Delta t = 10^{-2}$.*

potential), it was argued that the native state has a large basin of attraction [91]. The native confirmation can thus be accessed rapidly and over a rather wide temperature range and would appear to be kinetically two-state-like. For the later comparison, the β-type folded state of protein G is displayed in Figure 1 of Ref. [91]. All three neutral residues are concentrated in a turn region. The hydrophobic residues in the branches at either side of this turn tend to be in close contact with each other while the hydrophilic residues point outwards. We refer to this state as *native* in the remainder.

Parameters in the potential terms

The parameters in (4.16) are the same as in Ref. [91]: $k^b_{ij} = 100$, $d_0 = 1$, $k^a_{ijk} = 20$ and $\theta_0 = 105^\circ$ in the bond length and angle potentials, $A_i = 0$ and $B_i = 0.2$ if two or more of the four particles are neutral (N) and $A_i = B_i = 1.2$ for all other cases in the torsion potential (4.20). Non-bonded potentials are given by (4.21) with $\varepsilon_h = 1$ and coefficients $C_{ij} = -D_{ij} = \frac{2}{3}$ if one particle is of the L type and the other of type L or B, $C_{ij} = D_{ij} = 1$ if both particles are of type B and $C_{ij} = 0$ and $D_{ij} = -1$ if either one of two particles is of type N.

Distance measure

The distance between a simulated and a reference state is determined using the common structure overlap function χ,

$$\chi = 1 - \frac{2}{N^2 - 5N + 6} \sum_{i=1}^{N-3} \sum_{j=i+3}^{N} \Theta(\epsilon_{\text{tol}} - |d_{i,j} - d^{\text{ref}}_{i,j}|), \tag{4.22}$$

where N is the number of particles, d_{ij} and d^{ref}_{ij} are distances between particle i and j in the simulated and reference state, respectively, and Θ is the Heaviside function. The structure overlap function is insensitive to chain rotation or translation. If $|d_{i,j} - d^{\text{ref}}_{i,j}| \leq \epsilon_{\text{tol}}$, particles i and j are assumed to be in a contact according to the reference state. Unless mentioned otherwise, our reference state is the native state shown in Figure 4.8i. We use $\epsilon_{\text{tol}} = 0.2$ [91].

Regularization parameter and time step

Before reviewing simulation results, we shortly consider the regularization parameter and the time step. In all simulations with RFSU we used either $\varepsilon = 10^{-3}$ or 10^{-2}, a choice that is based on the results of our previous analysis. A large time step provides an enhanced sampling of the potential energy surface. Initially, we considered $\Delta t = 10^{-2}$ for RFSU and found that the simulations are stable. In CLD, such a large time step causes instability and we were forced to simulate with a reduced $\Delta t = 10^{-4}$. Nevertheless, this cause for instabilities is also active in the initial stages of RFSU, since the initial mobility $B_0 = I$ does not yet include proper scaling for the different modes. This phenomenon is clear in Figure 4.6a where one can observe irregular high peaks during the initial iterations of the simulation. In principle, one can remedy this behavior by preconditioning, i.e. determining a proper choice of $B_0 = J_0 J_0^T \neq I$. This approximate inverse Hessian B_0 can be determined by performing the S-QN method for $k_B T = 0$ (standard QN). Such a B_0 was found to enable stable simulation for an even increased $\Delta t = 10^{-1}$.

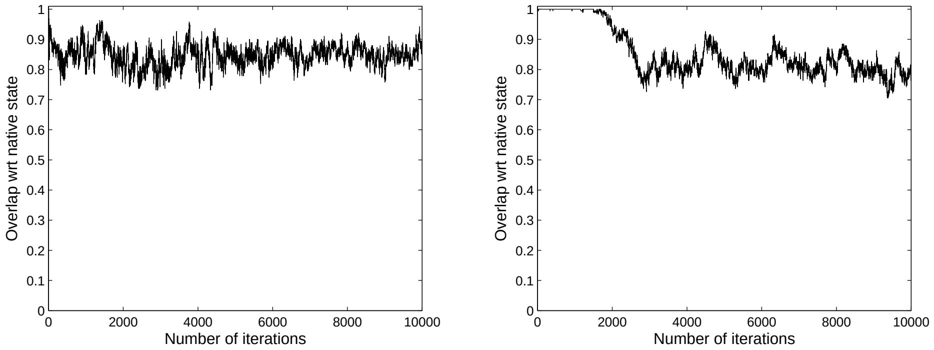
Starting configuration

The simulation pathway over the potential energy hypersurface naturally depends on its complex topography, making the observation of a folding event within the considered limits of simulation time sensitive to the starting state on this folding funnel. The starting state and influence of external factors in biological folding processes remain a matter of speculation, complicating the determination of realistic starting states. Many simulations of protein conformational changes or dynamics begin at the native crystal structure using the atomic coordinates measured from electron density maps obtained by X-ray diffraction methods. Other simulations start from more random conformations that often obey constraints, in that selected interatomic distances in the conformation have particular values. Sometimes this can be done analytically by choosing as random variables either the torsion angles, which always preserve bond length and bond angle values, or the rigid-body rotations and translations, which always preserve the internal structure of the rigid fragment. Methods for generating random structures that satisfy constraints on the distances between pairs of atoms or groups given upper and lower bounds also exist [110, 111]. Weak constraints on bond lengths, angles and torsions are also used in many coarse-grained descriptions, where random (coil) starting configurations are generated at elevated temperatures by pre-simulation, after disabling non-bonded interactions, or by simulated annealing (SA). The study of Veitshans [91] used a SA procedure to generate 100-300 independent states for protein G and showed that, for $T = 0.41 < T_F$, the fraction of unfolded chains decays roughly exponentially with time and vanishes around $t = 3000$. Here, we select specific starting configurations, the V-shape shown in Figure 4.8a, with upper and lower bounds for otherwise random bond length, angles and torsions. These structures are inspired by our focus on the collective dynamics and based on the identification of the turn region in the native state [91]. We expect that the domains at either sides of the turn point will collectively approach each other in RFSU to reach the native state. The large Φ of this state and the collective behavior that is required for folding suggest that this state is part of a region that is rather distant from the native state. Moreover, the fraction of folded proteins in CLD will depend on the cumulative Kramer's transitions rates for *all* potentials energy barriers between starting and native states. A direct comparison of our result to the folding rates obtained by Veitshans is therefore rather impossible [91]. Instead, we compare RFSU and CLD (with $B = \gamma I$) for the same (random) starting states.

Simulations for $T > T_\theta$

We start with simulations for $k_B T = 1$, which reduces to $T = 1$ in reduced units [91]. We note that this temperature is well above the collapse transition temperature $T_\theta = 0.78$. Our reason for this choice is that sampling at elevated temperatures increases the probability of barrier crossing. In Ref. [91], the averaged overlap function $\langle \chi \rangle$ was found to be $\langle \chi \rangle \approx 0.8$ for $T = 1$, where the averaging is carried out over pathways for different starting configurations. As mentioned, our starting configuration is the V-shaped conformation displayed in Figure 4.8a. For this structure, $\chi = 1$ and the domains at either side of the turning point are rather far apart. We used $B_0 = I$ and $\Delta t = 10^{-2}$ for S-QN, where the mobility B_{k+1} was determined by RFSU. For CLD, we used $\gamma = 1$ and $\Delta t = 10^{-4}$.

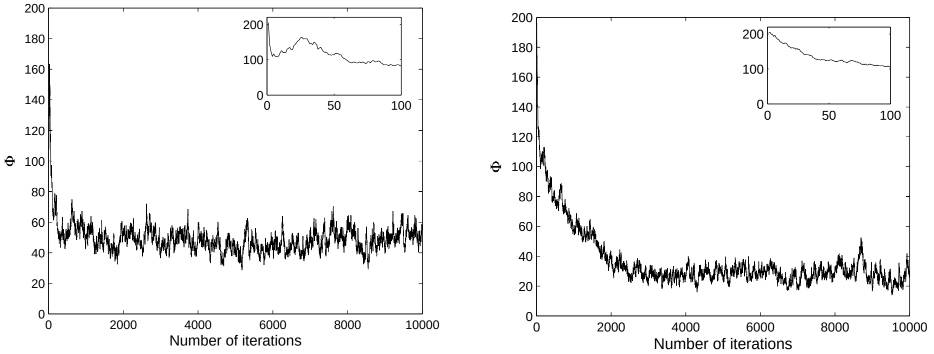
Figure 4.5 shows the evolution of the structure overlap function χ for both S-QN (a) and CLD (b). Although χ for S-QN is below the one for CLD for practically the complete range of time steps, their final (averaged) values are around the same $\chi \approx 0.8$, in good agreement the results of Veitshans *et al* [91]. Clearly, the chains exhibit in a random coil conformation during its evolution in both methods. The corresponding evolution of the potentials Φ for both simulations are shown in Figure 4.6. Although Φ falls off much faster for S-QN, it fluctuates at a somewhat higher range (values between 40 and 60) than the CLD simulations (between 20 and 40). Although this performance looks very similar, the information contained in Figures 4.5 and 4.6 is incomplete. To analyze the sampling in S-QN and CLD, we explored the potential well structure around the sampling path. In particular, we used conformations obtained at every 500 time steps in S-QN and CLD for quenching into local minima of Φ or *inherent structures* by a standard QN method with a backtracking line search. The rather flat wells and local minima of Φ are displayed in Figures 4.7a and 4.7b. Missing parts of the wells were added symmetrically. In Figures 4.7c (S-QN) and 4.7d (CLD) we show a correlation map of inherent states obtained for S-QN and CLD respectively. Using the overlap function χ as a measure for similarity between different inherent states, white squares denote fully uncorrelated structures ($\chi \approx 1$), while



(a) *Overlap function with respect to a configuration from the native class using the regularized FSU with* $\varepsilon = 10^{-3}$ *and*$\Delta t = 10^{-2}$.

(b) *Overlap function with respect to a configuration from the native class using the conventional Langevin dynamics with* $\Delta t = 10^{-4}$.

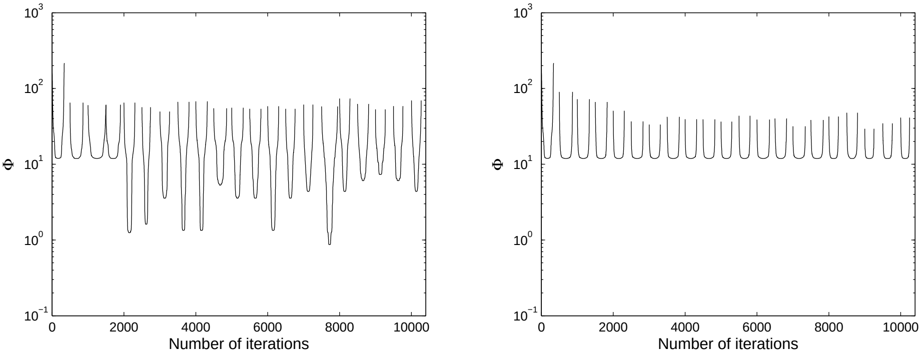
Figure 4.5: *Comparison of the structural overlap functions* χ *obtained by regularized FSU and standard Langevin dynamics for simulations of a model protein with* $k_B T = 1$.



(a) *Evolution of the potentials* Φ *obtained by the regularized FSU with* $\varepsilon = 10^{-3}$ *and*$\Delta t = 10^{-2}$.

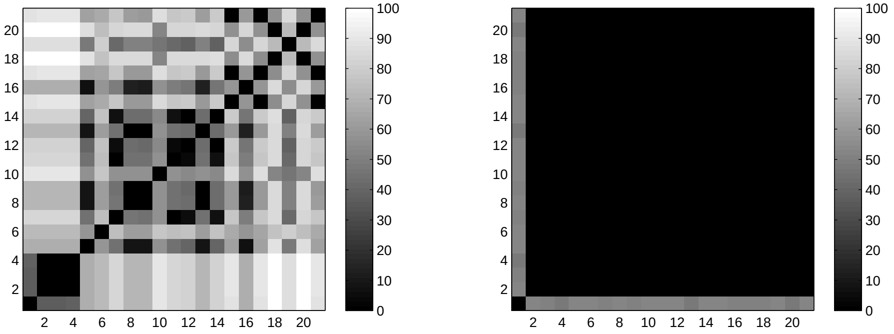
(b) *Evolution of the potentials* Φ *obtained by the conventional Langevin dynamics with* $\Delta t = 10^{-4}$.

Figure 4.6: *Evolution of the potentials* Φ *for simulations of a model protein with* $k_B T = 1$.



(a) *Local minima found during the simulation with regularized FSU.*

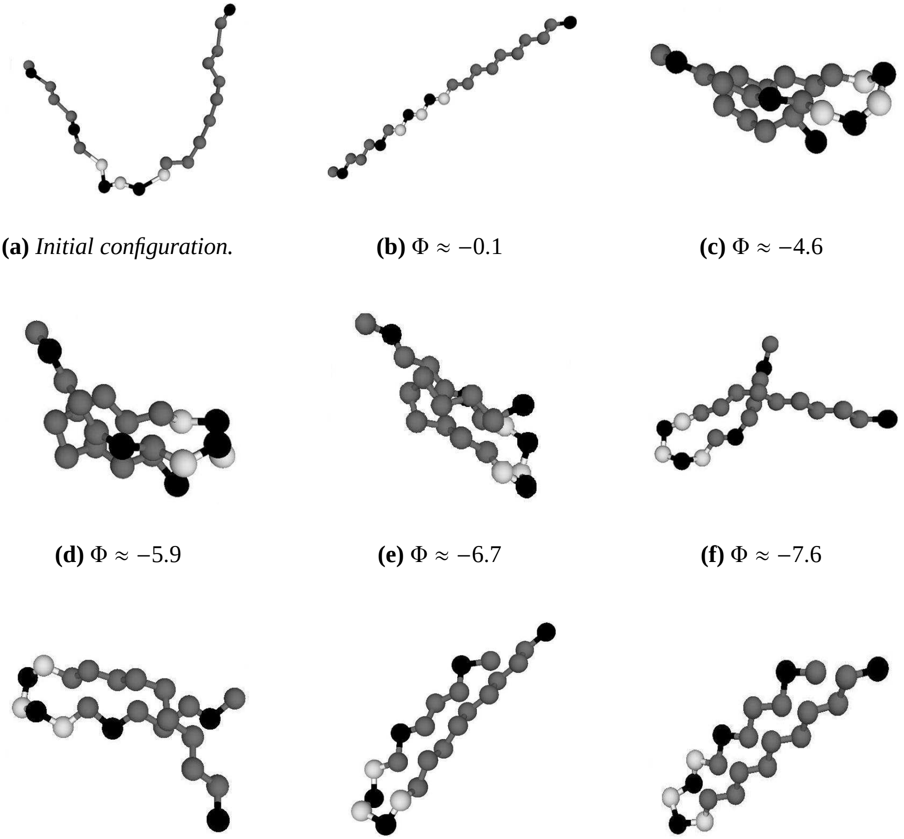
(b) *Local minima found during the simulation with standard Langevin dynamics.*



(c) *Correlations of the configurations corresponding to the local minima obtained during sampling with regularized FSU.*

(d) *Correlations of the configurations corresponding to the local minima obtained during sampling with standard Langevin dynamics.*

Figure 4.7: *Comparison of the local minima found by quenching after every 500 steps for the regularized FSU and the standard Langevin dynamics. In order to plot on a log-scale* Φ_+ *instead of* Φ *is shown, where* $\Phi_+ = \Phi + c_+$*, with* $c_+ = 12$*. The correlation matrices show the correlations between the configurations of the inherent state, the darker the square the more correlated the configurations are.*



(a) *Initial configuration.* **(b)** $\Phi \approx -0.1$ **(c)** $\Phi \approx -4.6$

(d) $\Phi \approx -5.9$ **(e)** $\Phi \approx -6.7$ **(f)** $\Phi \approx -7.6$

(g) $\Phi \approx -8.4$ **(h)** $\Phi \approx -10.4$ **(i)** $\Phi \leq -10.5$

Figure 4.8: *Configurations representing a typical structure from different classes found based on the correlation matrix.*

black squares denote fully correlated or similar structures ($\chi \approx 0$). The first entry in the correlation map is the starting structure shown in 4.8a and auto-correlation gives rise to a back diagonal.

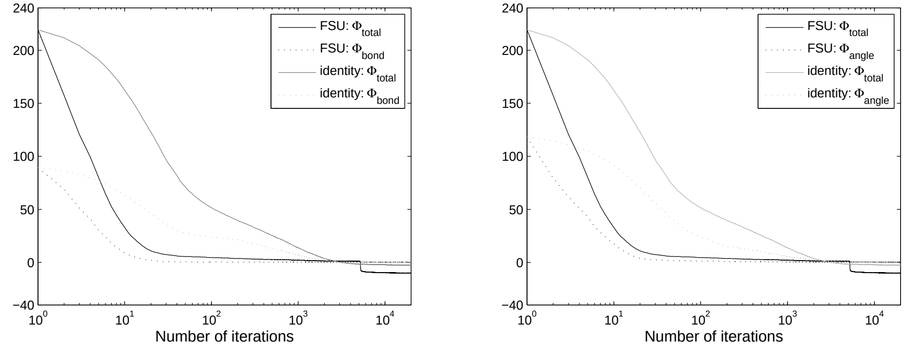
Correlated inherent structures were found to map one-to-one to distinct local minima of the potential energy Φ, ranging from $\Phi = -0.1$ to $\Phi = -11.1$. Based on this analysis, we classified the inherent structures for S-QN into several categories: $\Phi \approx -0.1$ ($k = 500, 1000, 1500$), $\Phi \approx -4.6$ ($k = 9000$), $\Phi \approx -5.9$

($k = 8500, 9500$), $\Phi \approx -6.7$ ($k = 4500$), $\Phi \approx -7.6$ ($k = 7000, 8000, 10000$), $\Phi \approx -8.4$ ($k = 3000, 5000, 5500, 6500$), $\Phi \approx -10.4$ ($k = 2500$) and $\Phi \leq -10.5$ ($k = 2000, 3500, 4000, 6000, 7500$). Representative structures are displayed in Figures 4.8b- 4.8i in the order of decreasing Φ. For CLD all local minima are the same, $\Phi \approx -0.1$ in Figure 4.7b, and only the inherent structure in Figure 4.8b was identified by the quenching procedure. These results show that a much larger part of the potential energy landscape is sampled by S-QN compared to CLD, which samples only very local, despite the apparently very similar χ in Figure 4.5. The structure in Figure 4.8i ($\Phi = -10.75$) is visually equal to the native structure of protein G in Figure 1 of Ref. [91]. We note that we identified a slightly different structure with even lower $\Phi = -11.13$. However, this structure belongs to the same class.

Simulations for $T < T_F$

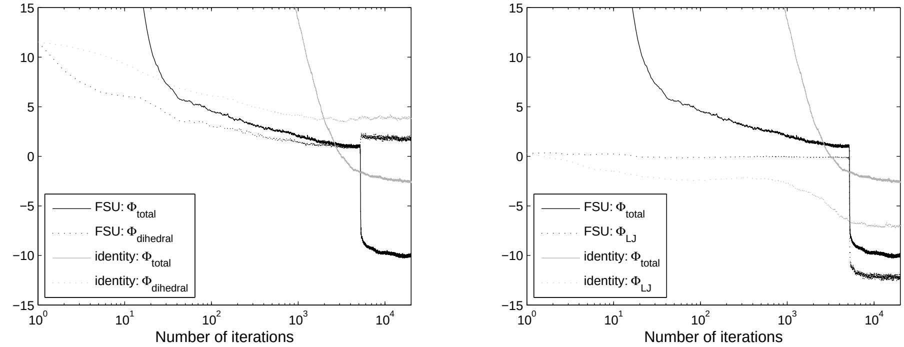
We focus on the multiscale nature of our method by considering a temperature $T = 0.01$, much lower than the folding temperature. For such a low temperature, folding may become very slow in conventional Langevin dynamics [91]. To anticipate, we selected several V-shaped random initial configurations that are not extremely far from the native state but still have minimal overlap: χ is close to one. We carried out a preconditioning step for the determination of B_0 in RFSU, which prevents the irregularity at the beginning of the simulation (see insert Figure 4.6a). This preconditioning enables stable simulations with $\Delta t = 10^{-1}$. For a genuine comparison of folding rates in S-QN and CLD, an appropriate constant mobility γ in CLD should be determined. We can either determine this mobility based on physical properties, i.e. the effective friction coefficient, or relate to the inverse Hessian in the initial $\mathbf{x}_0$. The fastest mode for a quadratic potential, the one that sets the maximum time step in CLD, is associated with the largest eigenvalue of H. Using the calculated largest eigenvalue $\lambda_{\max}(B_0^{-1}) \sim O(10^5)$ of the approximation B^{-1} of $H + \epsilon I \sim H$ in $\mathbf{x}_0$ and taking into account the time step in S-QN, we obtain an equivalent time step Δt for CLD of approximately 10^{-6}. Since we found the scheme to be stable for $\Delta t = 10^{-4}$, we use this value instead. For each method (S-QN and CLD) we performed 10+ simulations using different starting configurations. From each of these two sets, we selected and analyzed the one that reaches the lowest Φ.

From Figure 4.9a, we observe a much faster decrease of the potential energy Φ_{total} for S-QN than for CLD, despite the fact that we have chosen a larger time step in CLD. For better understanding of the folding kinetics, we considered the evolution of the different contributions to the energy potential separately (see Figures 4.9a, 4.9b, 4.9c



(a) *Comparision of the evolution of* Φ_{bond} *during simulations with the regularized FSU and the standard Langevin dynamisc.*

(b) *Comparision of the evolution of* Φ_{angle} *during simulations with the regularized FSU and the standard Langevin dynamisc.*



(c) *Comparision of the evolution of* Φ_{dihedral} *during simulations with the regularized FSU and the standard Langevin dynamisc.*

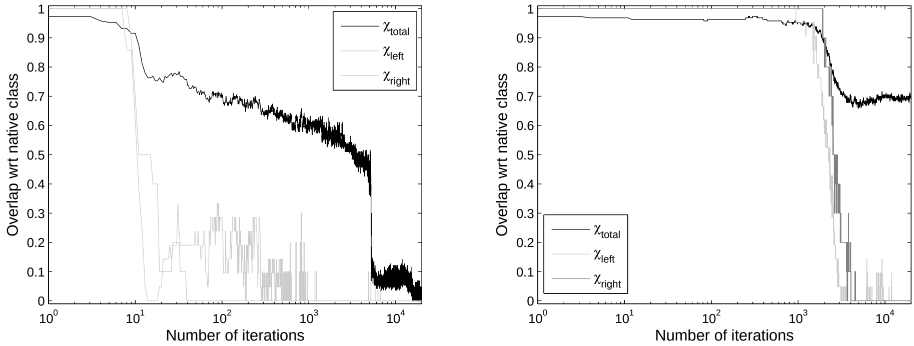
(d) *Comparision of the evolution of* Φ_{LJ} *during simulations with the regularized FSU and the standard Langevin dynamisc.*

Figure 4.9: *Comparision of the evolution of the different potentials during simulation with the regularized FSU and the standard Langevin dynamisc.*

and 4.9d). Collective modes in S-QN are typically first observed in the potentials for intra-chain interactions (Φ_{bond}, Φ_{bending} and Φ_{dihedral}) while the contribution of the non-bonded potential remains rather insignificant to later stages. The bond length and angles converge in $O(10)$ steps to their equilibrium values, see Figures 4.9a and 4.9b. From Figure 4.9c, one can observe that after $\sim$ 500 iterations the curves for Φ_{dihedral} and Φ_{total} have the same decreasing nature and eventually start to overlap after $k \sim 2000$. At this stage, the equilibration of the dihedral angles is clearly the main contribution to the decreasing potential Φ_{total}, an observation that is supported by the other intra-chain potentials fluctuating around zero. Following this stage, there is a sudden drop in Φ_{total} after 5000 iterations, reflecting an equivalent drop in Φ_{LJ}, Figure 4.9d. The torsion potential responds to this sudden drop by a small increase followed by equilibration. We conclude that this reflects a collective chain collapse where different chain domains at either side of the turn move coherently, a process that is considerably accelerated when the long-range Lennard-Jones interactions become more significant. In the rearrangement process that follows, most probably concentrated in the turn region, the bond lengths and bond angles remains constant but the torsion angles have to adapt.

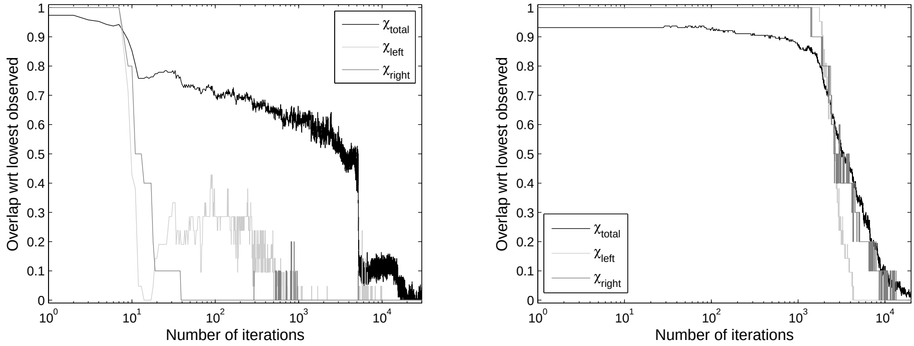
For CLD we see a gradual decrease of the intra-chain potentials and the non-bonded potentials, a signature of diffusion. The intra-chain potentials decrease much slower than for S-QN and the bonds and angles reach equilibrium values only after $O(10^3)$ steps. We note that large force constants in the bond terms, introduced by Veitshans *et al* [91] as an alternative to RATTLE or SHAKE, ensure some collectivity. Unlike S-QN, one can observe an immediate but small decrease of the non-bonded potential, Figure 4.9d, indicative of a very slow and sequential collapse of non-bonded interaction sites along the chain. From the values of torsion and non-bonded potentials at $k = 20000$ it is clear that this zippering-like process continues beyond the end of the simulation, and that the chain has not reached a stable state.

Next, we consider the structure overlap function $\chi = \chi_{\text{total}}$, using the structure of Figure 4.8i from the native class as a reference state. Although principal modes can in principle be determined as eigenvectors of B along the simulation pathway, domains that play a role in collective behavior can be anticipated from the start and the native state. We define partial overlap functions χ_{left} and χ_{right} that only consider two disjunct sub-domains LB_8 and BLB_3LB at either side of the turn region, respectively. We note that the partial overlap functions can both vanish even when the structure is not native. Sub-domains are chosen such that the turn region is excluded. In Figure 4.10a all structure overlap functions are combined for S-QN. It is clear that the left and the right domain assemble into a native conformation very fast. However, after χ_{left} and χ_{right} have vanished, they increase again at later stages and fluctuate between $0 - 0.3$ due to internal reorganizations. While the sub-domains rearrange (at least 70% stays conform to the native state), χ_{total} is gradually decreasing. After 1000 iterations, the rearrangements in the left and right domains damp out. The sudden drop in χ_{total} coincides with the drop in the non-bonded potential (see Figure 4.9d), when the LJ-



(a) *Evolution of the structural overlap with respect to a configuration from the native class as reference configuration during the simulation with regularized FSU.*

(b) *Evolution of the structural overlap with respect to a configuration from the native class as reference configuration during the simulation with standard Langevin dynamics.*



(c) *Evolution of the structural overlap with respect to the configuration corresponding to the lowest observed* Φ *during the simulation with regularized FSU.*

(d) *Evolution of the structural overlap with respect to the configuration corresponding to the lowest observed* Φ *during the simulation with standard Langevin dynamics.*

Figure 4.10: *Evolution of the structural overlap with respect to the configuration corresponding to the lowest observed* Φ *and with respect to the configuration from the native class during the simulation with regularized FSU and the standard Langevin equation.*

interactions become significant, and small rearrangements in subdomains. Once the native state is found (around $k = 15000$) only conformations in the native basin are sampled, since collective modes seize to contribute and the noise amplitude is low for $T = 0.01$. For a comparison, Figure 4.10b contains the overlap functions for conventional Langevin dynamics. It is clear that χ_{left} and χ_{right} decrease at least one order of magnitude slower than in S-QN. Although the sub-domains eventually reach their native conformation, rearrangement of the chain as a whole is much slower and the chain does not reach the native conformation within simulation time.

The analysis of the overlap function with respect to the native class seems arbitrary for comparing the dynamics, since conventional Langevin dynamics does not necessarily converge to the same class as S-QN. Therefore, we introduce different reference states, in particular the state with the lowest observed Φ during each of the simulations (the lowest observed state) by S-QN or CLD. In Figures 4.10c and 4.10c $\chi_{\text{total}}, \chi_{\text{left}}$ and χ_{right} for S-QN and CLD are shown, respectively. The analysis for the S-QN is very similar to the one discussed earlier, since the lowest observed state is in the class of native states. For CLD, we see that both sub-domains rearrange into their lowest observed structure very slowly and that the signature of χ_{total} is highly correlated with these partial overlap functions. In particular, the formation of optimal structure (almost) coincides with the formation of optimal partial structure in the two sub-domains. This is a clear sign of non-cooperative behavior. As a reference, partial structure forms at an earlier stage and moves collectively as the result of long range interactions in S-QN.

4.4. Discussion

Force constants

Fixed bond length are often enforced in molecular modeling by computationally demanding SHAKE or RATTLE algorithms [107, 108]. An alternative constraint method, LINCS (Linear Constraint Solver) developed in 1997 [109], directly resets the constraints rather than the derivatives of the constraints and is up to four times faster than SHAKE. In Ref. [91], an alternative approach is employed to circumvent the computational demands associated with these methods. Stiff bonds are enforced by setting the force constants to a high $k = 100$. A trade-off of this approach is that the time step should be reduced. The automated scaling provided by S-QN gives rise to a fast convergence to the equilibrium chain length l_{eq} and 'soft' constraints on the bond length, as we showed for Rouse chains in one and three dimensions, even for a very small force constant $k = 1$. From this protein study, we found that the minimization of the intra-chain energy potentials, and also Φ itself, is rather insensitive to a scaling of the potential

$$\Phi_\alpha = \alpha(\Phi_{\text{bond}} + \Phi_{\text{bending}} + \Phi_{\text{dihedral}}) + \Phi_{\text{non}}, \tag{4.23}$$

where α was varied between 0.1 and 1. Although they may vary in details, the signatures of both the intra-molecular potentials and the (partial) overlap functions do not depend on the considered values of α. In particular, the overlap functions is of the same form as illustrated in Figure (4.10a): first the partial structures (left and right part) are established, followed by a drop in χ_{total} indicating the increasing influence of long range interactions. Hence, the bond length is *automatically* constrained and the force constants in the potential can be significantly reduced. This observation provides tools for further optimization of performance.

Different starting configurations and regularization parameter choice

We accounted for the kinetic arrest that may occur at a very low temperature $k_BT = 0.01$ by considering a sharper V-shaped starting conformation than shown in Figure 4.8a for $k_BT = 1$. Moreover, the regularization parameter ε used in RFSU for $k_BT = 0.01$ is one order of magnitude larger ($\varepsilon = 10^{-2}$) than in the simulation with $k_BT = 1$ ($\varepsilon = 10^{-3}$) to avoid over-acceleration. Since the value of regularization parameter ε determines the upper bound to the acceleration of the collective modes, i.e. the damping with respect to the non-regularized case, a reduced ε speeds up the slowest collective modes that are important for bringing the structure of Figure 4.8a to the native configuration. We have also performed simulations with V-shaped starting configurations that have minimal overlap at the beginning of the simulations, $\chi \approx 1$, and found very similar signatures in the overlap functions and the contribution of the potential terms.

Partial structures

The partial structures LB_8 and BLB_3LB, important for the analysis of collective modes, were chosen based on the native chain conformation shown in Figure 4.8i, or alternatively Figure 1 of Ref. [91]. The question may arise whether our analysis is sensitive to this particular choice. For this reason, we systematically varied the partial structures. The evolution of the partial overlap functions for the left wing, LB_i with $i = 3..7$, converges to zero in the same way as for LB_8 in Figure 4.10a. These partial structures, however, remain in their native configuration after this stage, unlike LB_8 that was shown to fluctuate between 0 and 0.3. For the right partial structure, we also calculated the partial overlap function for LB_3LB, B_3LB and B_2LB. Shortening the sub-domain leads to increased convergence to the native partial structure, although all convergence rates are very similar to the one for BLB_3LB ($O(10)$). Extending partial structures beyond the chosen ones affects the evolution of overlap functions. The fluctuations in the overlap function of LB_8 are caused by one of the B beads, that is part of the turn region. Including the turn region completely, by dividing the chain in two equally large sub-domains LB_9N and $(LN)_2BLB_3LB$, slows down the convergence rates of the partial overlap functions, since the turn region has to facilitate collective rotation and contraction by local rearrangements. Nevertheless, for RFSU these partial overlap functions posses the same features as in Figure (4.10a): the partial overlap functions vanish before the chain as a whole finds it native state.

4.5. Conclusions

We applied a regularized stochastic Quasi-Newton method to study a protein model. The new form of regularization was incorporated in the factorized update (FSU) algorithm used for the determination of the curvature-dependent mobility B in S-QN and resolves often occurring problems associated with a singular or ill-conditioned local curvature H. The regularized FSU (RFSU) scheme is based on determining an estimate of local curvature (B^{-1}) that converges to $H + \varepsilon I$ instead of H, and does not affect the efficiency of the original FSU scheme [81].

This work is part of a series of articles towards the development of a general efficient and stable method for thermodynamically consistent accelerated equilibration. In Ref. [80], we introduced the principles of S-QN and proved enhanced sampling performance and barrier crossing for 1-and 2-dimensional examples. In the original S-QN, standard Davidon-Fletcher-Powell (DFP) and Choleski decomposition was used for the determination of B and the noise amplitude, respectively. In Ref. [81], we introduced cost effective FSU and limited-memory (L-)FSU algorithms for determining L with $B = LL^T$ and analyzed the sampling performance and multi-scale nature for a simple but physically relevant Rouse chain with quadratic potential. Here, we regularized S-QN via RFSU and applied the resulting method for a previously developed coarse-grained model of a rather short protein with a β-type folded state. We developed an efficient two-step approach: the enhanced sampling gives rise to many inherent states, i.e. local minima on the energy landscape, when simulating above the collapse temperature, and, below the folding temperature, the cooperative modes of a protein can be efficiently determined and considerably speed up the folding process when compared to conventional Langevin dynamics.

A detailed analysis of the S-QN results shows that the folding pathway can be divided in local and collective parts. Bond length, bond angles and torsions are equilibrated in this order, a feature that is rather insensitive to the force constants used in the potentials. After this equilibrium is established, a sudden drop in the total potential occurs, corresponding to the non-bonded interactions becoming more significant. This sudden drop coincides with the sudden drop in the total overlap function, that provides a distance measure with respect to the native state. The overlap function also shows that partial structures, i.e. carefully chosen sub-domains of the protein, form prior to this drop and move collectively to reach the native state. This collective dynamics is absent in conventional Langevin dynamics, where native contacts form as a result of (local) diffusion.

The considered coarse-grained protein is rather short and was previously determined to be a good folder [91]. Moreover, the native structure was known from this earlier study, although we identified a slightly different structure with even lower potential energy. It is tempting to use the two-step approach for the determination of inherent structure and principle mode analysis of proteins with direct biological function. Crystal structures can serve as input for the determination of inherent structures of biological relevance, and inherent structures can be analyzed for the class of proteins that are not easily crystallized, for instance membrane proteins. Realistic coarse-grained parameters for these proteins can in principle be determined by systematic coarse-graining procedures and external factors (for instance, chaperones) can be explicitly included in the S-QN method. We leave such study for future work.

CHAPTER 5

Efficient calculation of the generalised Langevin equation

In the previous chapters, we have introduced our S-QN method. Since the calculation of the curvature-dependent mobility and its factorized form is very compute expensive, we also introduced an efficient factorized update scheme. The factorized update scheme (FSU) enables direct updates of the noise term and therefore avoids the compute expensive (Cholesky) factorization. In the previous chapter, where we applied S-QN to a minimal protein model, the small number of degrees of freedom (n) rendered FSU appropriate for the calculation of the mobility matrix. For very large systems, however, storage and modification of full matrices becomes inefficient and L-FSU is more appropriate. Nevertheless, in Chapter 3 we have only considered L-FSU for a quadratic potential, i.e. the case where the spurious drift term vanishes. In this chapter, we clarify how to efficiently combine L-FSU and the scheme introduced by Hütter and Öttinger [39], that applies when the spurious drift term becomes significant and involves the calculation of the inverse of B.

5.1. Introduction

Our starting point is the generalized Langevin equation:

$$\mathrm{d}\mathbf{x} = [-B(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T\nabla \cdot B(\mathbf{x})]\mathrm{d}t + \sqrt{2k_B T}\, J(\mathbf{x})\mathrm{d}W(t), \tag{5.1}$$

where $B(\mathbf{x}) = J(\mathbf{x})J(\mathbf{x})^T$. It is clear that factorization of $B(\mathbf{x})$ and calculation of $\nabla \cdot B(\mathbf{x})$ are of the highest computational complexity.

Avoid factorizing the mobility

Using our proposed factorized secant update (FSU) scheme from Chapter 3, a direct update of J is possible;

$$J_{k+1} = J_k + \frac{\alpha \mathbf{s}_k \mathbf{y}_k^T J_k - \alpha^2 J_k J_k^T \mathbf{y}_k \mathbf{y}_k^T J_k}{\mathbf{y}_k^T \mathbf{s}_k}, \tag{5.2}$$

where $\mathbf{s}_k = \mathbf{x}_{k-1} - \mathbf{x}_k$, $\mathbf{y}_k = \nabla\Phi(\mathbf{x}_{k-1}) - \nabla\Phi(\mathbf{x}_k)$ and

$$\alpha^2 = \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{y}_k^T J_k J_k^T \mathbf{y}_k}. \tag{5.3}$$

Avoid calculating the divergence of the mobility

Using the scheme of Hütter and Öttinger [39], no explicit calculation of the divergence of the mobility is needed;

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{x}_k + \Delta\mathbf{x}_k && (5.4)\\
\Delta\mathbf{x}_k &= -\frac{1}{2}[B(\mathbf{x}_k + \Delta\mathbf{x}_k^p)\nabla\Phi(\mathbf{x}_k + \Delta\mathbf{x}_k^p) + B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)]\Delta t \\
&+ \frac{1}{2}[B(\mathbf{x}_k + \Delta\mathbf{x}_k^p)B^{-1}(\mathbf{x}_k) + I]\sqrt{2k_B T}\, J(\mathbf{x}_k)\Delta W_t, && (5.5)\\
\Delta\mathbf{x}_k^p &= -B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\Delta t + \sqrt{2k_B T}\, J(\mathbf{x}_k)\Delta W_t. && (5.6)
\end{aligned}$$

where (5.6) is the predictor step and (5.4) is the correction. Although this predictor-correction scheme avoids the calculation of the divergence of B, other costs are involved, notably the significant costs for the calculation of the inverse.

Fortunately the DFP update scheme (equivalent to FSU) for B_k is of such a form that the Sherman-Morrison theorem can be applied to calculate the exact inverse of $B^{-1}(\mathbf{x}_k) = G(\mathbf{x}_k) = G_k$ in (5.5)

$$G_k = (I - \frac{\mathbf{y}_{k-1}\mathbf{s}_{k-1}^T}{\mathbf{y}_{k-1}^T\mathbf{s}_{k-1}})G_{k-1}(I - \frac{\mathbf{s}_{k-1}\mathbf{y}_{k-1}^T}{\mathbf{y}_{k-1}^T\mathbf{s}_{k-1}}) + \frac{\mathbf{y}_{k-1}\mathbf{y}_{k-1}^T}{\mathbf{y}_{k-1}^T\mathbf{s}_{k-1}}. \tag{5.7}$$

Using FSU and the update scheme of Hütter and Öttinger together with (5.7) decreases the computational cost from $O(n^3)$ to $O(n^2)$.

5.2. Theory

The Sherman-Morrison formula is a special form of the Woodbury formula for the inverse of a rank k correction matrix and is given by

$$\left[A + \mathbf{u}\mathbf{v}^T\right]^{-1} = A^{-1} - \frac{A^{-1}\mathbf{u}\mathbf{v}^T A^{-1}}{1 + \mathbf{v}^T A^{-1}\mathbf{u}}, \tag{5.8}$$

for an invertible square matrix A and vectors $\mathbf{u}$ and $\mathbf{v}$. Since the update scheme (5.2) is of the same form as the left hand side of equation (5.8), the inverse of J_{k+1} can be written as

$$J_{k+1}^{-1} = F_{k+1} = F_k - F_k\frac{\mathbf{s}_k - \alpha_k J_k J_k^T\mathbf{y}_k}{\mathbf{y}_k^T\mathbf{s}_k}\mathbf{y}_k^T, \tag{5.9}$$

where $J_k^{-1} = F_k$. It is easily checked that $J_{k+1}F_{k+1} = I$. In equation (5.5) only B^{-1} occurs and the factorized form of B^{-1} seems redundant. However deriving F_k enables us to rewrite B^{-1} in a suitable way for a limited memory construction of B^{-1}. Rewrite $B^{-1}(\mathbf{x}_{k+1})$ as

$$F_{k+1}^T F_{k+1} = W_k^T W_{k-1}^T ... W_0^T F_0^T F_0 W_0 ... W_{k-1} W_k, \tag{5.10}$$

with

$$\begin{aligned} W_k &= (I - \frac{\mathbf{s}_k - \alpha_k J_k J_k^T\mathbf{y}_k}{\mathbf{y}_k^T\mathbf{s}_k}\mathbf{y}_k^T) = (I - \frac{\mathbf{s}_k - \alpha_k\mathbf{h}_k}{\mathbf{y}_k^T\mathbf{s}_k}\mathbf{y}_k^T) & (5.11)\\ &= (I + \frac{1}{\alpha_k\nu_k}\mathbf{v}_k\mathbf{y}_k^T), & (5.12)\end{aligned}$$

where $\mathbf{v}_k = \mathbf{h}_k - \mathbf{s}_k/\alpha_k$, $\mathbf{h}_k = J_k J_k^T\mathbf{y}_k$ and $\nu_k = \mathbf{h}_k^T\mathbf{y}_k$. The inverse of the mobility is now casted in a factorized form (5.10) where a recursive expression (obtained by loop unrolling), can serve as a basis for limited-memory implementation. Since the case for $k < m$ is trivial, we only consider the case where k is larger than or equal to the truncation parameter m. Following the derivation of L-FSU in appendix B, we derived the limited-memory case of B_{k+1}^{-1} for $k \geq m$

$$F_{k+1}^T F_{k+1} \;=\; \tilde{W}_k^T \tilde{W}_{k-1}^T \;\;\dots\;\; \tilde{W}_{k-m+1} F_0 F_0^T \tilde{W}_{k-m+1} \;\;\dots\;\; \tilde{W}_{k-1}\tilde{W}_k. \tag{5.13}$$

The ˜ above W_k indicates that W_k is defined as in equation (5.11) with truncated J_k, i.e.

$$J_k = \tilde{V}_{k-1}...\tilde{V}_{k-m+1}J_0, \tag{5.14}$$

where $\tilde{V}_k$ is defined as

$$\tilde{V}_k = (I - \frac{1}{\tilde{\mathbf{h}}_k^T\mathbf{y}_k}(\tilde{\mathbf{h}}_k - \frac{\mathbf{s}_k}{\tilde{\alpha}_k})\mathbf{y}_k^T), \tag{5.15}$$

with truncated J_k in $\tilde{\mathbf{h}}_k$ and $\tilde{\alpha}_k$.

Alternatively, the limited memory case of B_{k+1}^{-1} can be found by solving the rewritten $B_{k+1}B_{k+1}^{-1} = I$ as

$$[J_0^T(\prod_{i=1}^{m}\tilde{V}_{k-m+i}^T)]^T J_0^T(\prod_{i=1}^{m}\tilde{V}_{k-m+i}^T)[Z_0(\prod_{i=1}^{m}\tilde{Z}_{k-m+i})]^T Z_0(\prod_{i=1}^{m}\tilde{Z}_{k-m+i}) = I, \tag{5.16}$$

where $\tilde{V}_i$ is given in (5.15). A solution for equation (5.16) is found by solving $\tilde{Z}_k$ in

$$\tilde{V}_k^{-1} = \tilde{Z}_k, \tag{5.17}$$

and set $Z_0^T = J_0^{-T}$. Equation (5.17) can easily solved by applying the Sherman-Morrison formula on (5.15) which gives

$$\begin{aligned}
\tilde{Z}_k &= (I - \frac{1}{\tilde{\mathbf{h}}_k^T\mathbf{y}_k}(\tilde{\mathbf{h}}_k - \frac{\mathbf{s}_k}{\tilde{\alpha}_k})\mathbf{y}_k^T)^{-1} \\
&= I - \frac{-1}{\tilde{\mathbf{h}}_k^T\mathbf{y}_k}(\tilde{\mathbf{h}}_k - \frac{\mathbf{s}_k}{\tilde{\alpha}_k})\mathbf{y}_k^T/(1 + \frac{-\mathbf{y}_k^T}{\tilde{\mathbf{h}}_k^T\mathbf{y}_k}(\tilde{\mathbf{h}}_k - \frac{\mathbf{s}_k}{\tilde{\alpha}_k})) = I + \frac{1}{\tilde{\mathbf{h}}_k^T\mathbf{y}_k}(\tilde{\mathbf{h}}_k - \frac{\mathbf{s}_k}{\tilde{\alpha}_k})\mathbf{y}_k^T/\tilde{\alpha}_k \\
&= I + \frac{1}{\alpha_k\nu_k}\mathbf{v}_k\mathbf{y}_k^T.
\end{aligned} \tag{5.18}$$

Evidently (5.18) is the same as the derived $\tilde{W}_k$ from the truncated $F_{k+1}^T F_{k+1}$.

5.3. Discussion and Conclusion

Having derived a limited memory scheme for B^{-1}, we are now able to calculate the general Langevin equation in a complete limited memory way. Using the limited memory method will not only limit the storage needed, but the computational cost will decrease significantly. We finalize this chapter by considering the total computational cost of the general Langevin equation with adaptive mobility using the limited memory method. Only considering the computational load of the displacement (5.5) is sufficient since the predictor term (5.6) also appears in the displacement and thus reusable. Written out the discretized displacement gives

$$\begin{aligned}\Delta\mathbf{x}_k \quad = & \\ & -\frac{1}{2}B(\mathbf{x}_k^p)\nabla\Phi(\mathbf{x}_k^p)\Delta t && \text{(5.19a)}\\ & -\frac{1}{2}B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\Delta t && \text{(5.19b)}\\ & +\frac{1}{2}B(\mathbf{x}_k^p)B^{-1}(\mathbf{x}_k)\sqrt{2k_BT}\,J(\mathbf{x}_k)\Delta W_t && \text{(5.19c)}\\ & +\frac{1}{2}\sqrt{2k_BT}\,J(\mathbf{x}_k)\Delta W_t, && \text{(5.19d)}\end{aligned}$$

where $\mathbf{x}_k^p = \mathbf{x}_k + \Delta\mathbf{x}_k^p$. Terms (5.19b) and (5.19d) has already been discussed in the previous chapter: the computational costs are $10mn + 2n$ and $2mn + n$ respectively, where m is the truncation parameter or history depth and n the system size (dimension). The computational cost of term (5.19b) is an addition of the cost for the force times the limited factorized $B(\mathbf{x}_k)$ (5mn+n) and the cost for calculating $\mathbf{h}_k$, needed for the next time step. The computational cost for term (5.19a) is $5mn + n$ since it can be calculated in the same way as for (5.19b). Rest us to calculate the cost for term (5.19c). Since the vector $\sqrt{2k_BT}\,J(\mathbf{x}_k)\Delta W_t$ has already been calculated, we concentrate on the calculation of the vector times the limited factorized $B(\mathbf{x}_k)^{-1}$, which can be casted in the following scheme

$$\mathbf{d} = \sqrt{2k_BT}\,J(\mathbf{x}_k)\Delta W_t; \tag{5.20}$$

$$\begin{cases} \textbf{for } i = k-1, ..., \max(0, k-m) \\ \quad \gamma_i = \mathbf{y}_i^T \mathbf{d}; \\ \quad \mathbf{v}_i = \mathbf{h}_i - \mathbf{s}_i/\alpha_i; \\ \quad \mathbf{d} = \mathbf{d} + (\gamma_i/\alpha_i \nu_i)\mathbf{v}_i; \\ \textbf{end} \end{cases} \tag{5.21}$$

$$\mathbf{d} = F_0^T F_0 \mathbf{d}; \tag{5.22}$$

$$\begin{cases} \textbf{for } i = \max(0, k-m), ..., k-1 \\ \quad \omega_i = \mathbf{v}_i^T \mathbf{d}; \\ \quad \mathbf{d} = \mathbf{d} + (\omega_i/\alpha_i \nu_i)\mathbf{y}_i; \\ \textbf{end} \end{cases} \tag{5.23}$$

$$\textbf{stop with result } \mathbf{d} = F(\mathbf{x}_k)^T F(\mathbf{x}_k)\mathbf{d} = B(\mathbf{x}_k)^{-1} \sqrt{2k_B T} J(\mathbf{x}_k)\Delta W_t, \tag{5.24}$$

where the ˜ has been omitted for simpler notation. The first loop recursion (5.21) and second loop recursion (5.23) require $3mn$ and $2mn$ multiplications respectively; if $F_0^T F_0$ is diagonal, then n additional multiplications are needed. The vector (5.24) can be used to multiply with the limited version of $B(\mathbf{x}_k + \Delta\mathbf{x}_k^p)$, which are an additional $5mn + n$ multiplications. This gives us a total of $10mn + 2n$ multiplications for term (5.19c). The computational costs are summarized and compared in the table below. Clearly the computational complexity of the limited memory method is reduced significantly compared to the conventional calculation methods and FSU.

scheme term	Conventional		FSU		L-FSU	
	operation	cost	operation	cost	operation	cost
$J(\mathbf{x}_k)J(\mathbf{x}_k)^T\mathbf{y}_k$	NA		NA		store as $\mathbf{h}_k$ $\left[JJ^T\right]_{\text{lim}}\mathbf{v}$	$5mn+n$
$B(\mathbf{x}_k^p)\nabla\Phi(\mathbf{x}_k^p)$	$B_{\text{DFP}}\mathbf{v}$	$2n^2+O(n)$	$B_{\text{DFP}}\mathbf{v}$	$2n^2+O(n)$	$\left[JJ^T\right]_{\text{lim}}\mathbf{v}$	$5mn+n$
$B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)$	update $B(\mathbf{x}_k)$ $B_{\text{DFP}}\mathbf{v}$	$2n^2+O(n)$ $n^2+O(n)$	update $B(\mathbf{x}_k)$ $B_{\text{DFP}}\mathbf{v}$	$2n^2+O(n)$ $n^2+O(n)$	store $\mathbf{y}_{k-1}$,$\mathbf{s}_{k-1}$ $\left[JJ^T\right]_{\text{lim}}\mathbf{v}$	$5mn+n$
$J(\mathbf{x}_k)\Delta W_t(=\mathbf{w})$	Chol($B(\mathbf{x}_k)$) $J_{\text{Chol}}\mathbf{v}$	$O(n^3)$ $n^2+O(n)$	update $J(\mathbf{x}_k)$ $J_{\text{FSU}}\mathbf{v}$	$4n^2+O(n)$ $n^2+O(n)$	$\left[JJ^T\right]_{\text{lim}}\mathbf{v}$	$2mn+n$
$B(\mathbf{x}_k^p)B^{-1}\mathbf{w}$	inv($B(\mathbf{x}_k)$) $B_{\text{DFP}}B^{-1}\mathbf{v}$	$O(n^3)$ $3n^2+O(n)$	SM($B(\mathbf{x}_k)$) $B_{\text{DFP}}B^{-1}\mathbf{v}$	$2\frac{1}{2}n^2+O(n)$ $3n^2+O(n)$	$\left[JJ^TF^TF\right]_{\text{lim}}\mathbf{v}$	$10mn+n$

$B_{xxx}\mathbf{v}$ is the matrix vector multiplication where B is obtained by applying scheme xxx
Chol(B) is the Cholesky decomposition applied on B
inv(B) is the inverse calculation(Gaussian elimination) for the matrix B,
SM(B) is the inverse calculation using the Sherman−Morrison formula for the matrix B
$[..]_{\text{lim}}\,\mathbf{v}$ is the vector obtained using the limited memory methods

Table 5.1: *Computational cost comparison using different schemes.*

Reducing computational complexity and saving on storage is very important for simulating large systems. In this chapter we achieved to construct a limited-memory scheme for the generalized S-QN method. Combining the limited memory scheme with the automated multi-scaling property of the alternative mobility gives us a very powerful method for configurational space sampling with good thermodynamical consistency.

Bibliography

[1] E. Nelson, Dynamical Theories of Brownian Motion, Princeton University Press, Princeton, NJ (1967).

[2] R. Brown, A brief Account of Microscopical Observations made in the Months of June, July, and August, 1827, on the Particles contained in the Pollen of Plants; and on the general Existence of active Molecules in Organic and Inorganic Bodies, Philosophical Magazine N. S. **4**, (1828).

[3] R. Brown, Additional Remarks on Active Molecules, Philosophical Magazine N. S. **6**, (1829).

[4] A. Einstein, "Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen.", Annalen der Physik **17** (1905).

[5] A. Einstein, "Eine neue Bestimmung der Moleküldimensionen.", Inaugural-Dissertation zur Erlangung der philosophischen Doctorwürde der hohen Philosophischen Fakultät (Mathematisch-Naturwissenschaftliche Sektion) der Universität Zürich,K.J. Wyss, (Bern, 1905).

[6] M. Smoluchowski, "Zur kinetischen Theorie der Brownschen Molekularbewegung und der Suspensionen", Annalen der Physik **21**, (1906).

[7] P. Langevin, "Sur la theorie du mouvement brownien", C.R. Acad. Sci. **146**, (1908).

[8] J. Perrin, a number of short contributions between May 1908 and September 1909, C.R. Acad. Sci., (1908-1909).

[9] C.W. Gardiner. Handbook of Stochastic Methods. Springer-Verlag, Berlin, New York (1983).

[10] L. Verlet, Phys. Rev. **159**, 98 (1967).

[11] L. Verlet, Phys. Rev. **165**, 201 (1967).

[12] R. P. Feynman, R. B. Leighton and M. Sands, The Feynman Lectures on Physics, Vol. 1, Addison-Wesley, (1963).

[13] D. Frenkel and B. Smit, Understanding molecular simulation: from algorithms to applications. Academic Press, (1996).

[14] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.N. Teller and E. Teller, J. Chem. Phys. **21**, 1087 (1953).

[15] C. Pangali, M. Rao and B.J. Berne, Chem. Phys. Lett. **55**, 413 (1978).

[16] P.J. Rossky, J.D. Doll and H.L. Friedman, J. Chem. Phys. **69**, 4628 (1978).

[17] D. L. Ermak, J. Chem. Phys. **62**, 4189 (1975).

[18] S. Kirkpatrick, C. Gelatt and M. Vecchi, Science **220**,671 (1983).

[19] V. Cerny, J. Optim. Theory Applic. **45**, 41 (1985)

[20] P. J. M. V. Laarhoven and E. H. L. Aarts, Simulated Annealing: Theory and Applications, D. Rediel, Dordrecht (1987).

[21] J. Nocedal and S.J. Wright. Numerical Optimisation, Springer Series in Operation Research, New York (1999).

[22] R. M. Levy, A. R. Srinivasan, W. K. Olson, and J. A. McCammon, Biopoly- mers **23**, 1099 (1984).

[23] A. E. García, Phys. Rev. Lett., **68**, 2696 (1992).

[24] A. Amadei, A. B. M. Linssen and H. J. C. Berendsen, Struct. Funct. Gen. **17**, 412 (1993).

[25] F.H. Stillinger, Science **267**, 1935 (1996).

[26] F.H. Stillinger and T.A. Weber, Science **225**, 983 (1984).

[27] W.F. van Gunsteren and H.J.C. Berendsen, Molecular Physics **45**, 637 (1982).

[28] I.M de Schepper and E.G.D. Cohen, J. Stat. Phys. **27**, 223 (1982).

[29] E. Leontidis and U.W. Suter, Molecular Physics **83**, 489 (1994).

[30] S. Goldman, J. Chem. Phys. **62**, 441 (1986).

[31] M. Doi, S.F. EDWARDS, The theory of polymer dynamics, Oxford science publications, Oxford (1986).

[32] L. Kaufman, SIAM J. Optim. **10**, 56 (1999).

[33] J.E. Dennis Jr. and R.B. Schnabel, Numerical Methods for Unconstrained Optimisation and Nonlinear Equations, Siam Classics, (1996)

[34] M.J.D. Powell. In: Nonlinear programming, SIAM-AMS proceedings vol. IX, eds. R.W. Cottle and C.E. Lemke, SIAM Publications, Philadelphia (1976).

[35] J.M. Anglada and J.M. Bofill, J. Comput. Chem. **19**, 349 (1998).

[36] H.P. Hratchian and H.B. Schlegel. J. Chem. Theory & Comput. **1**, 61 (2005).

[37] P.E. Gill, W. Murray. J. Inst. Maths. Applns. **9**, 91 (1972).

[38] Grandinetti. In: Methods for operations research, Hain Verlay, (1979).

[39] M. Hütter and H.C. Öttinger, J. Chem. Soc., Faraday Trans., **94**(10), 1403 (1998).

[40] H.D. Vollmer, H. Risken. Z. Physik B. **34**, 313 (1979).

[41] C.W. Gardiner, Journal of Statistical Physics, **30**, No 1, 157 (1983).

[42] H.A. Kramers, Physica, **7**, 284 (1940).

[43] C.J. Cerjan and W.H. Miller, J. Chem. Phys. **75**, 2800 (1981).

[44] B.A. Berg and T. Neuhaus. Phys. Rev. Lett. **68**, 9 (1992).

[45] E. Aarts and J. Korst. Simulated Annealing and Boltzmann Machines. Wiley, Great

Britain (1989).

[46] C.D. Chau, G.J.A. Sevink and J.G.E.M. Fraaije, J. Chem. Phys. **128**, 244110 (2008).

[47] B. Dünweg, *Accelerated algorithms 2, Computer simulations of surfaces and interfaces, proceedings of the NATO Advanced Study Institute*, edited by B. Dünweg, D. P. Landau, A. Milchev (Kluwer Academic Publishers, Dordrecht / Boston / London 2003).

[48] U.H.E. Hansmann, Chem. Phys. Lett. **281**, 140 (1997).

[49] Y. Sugita and Y. Okamoto, Chem. Phys. Lett. **314**, 141 (1999).

[50] E. Marinari and G. Parisi, Europhys. Lett. **19**, 451 (1992).

[51] P. Liu, B. Kim, R. A. Friesner and B. J. Berne, Proc. Nat. Acad. Sci. USA. **102**, 13749 (2005).

[52] N. Nakajima, J. Higo, A. Kidera and H. Nakamura, Chem. Phys. Lett. **278**, 297 (1997).

[53] F. Wang and D. P. Landau, Phys. Rev. Lett. **86**, 2050 (2001).

[54] A. Laio, M. Parrinello, Proc. Natl. Acad. Sci. USA. **99**, 12562 (2002).

[55] A. F. Voter, J. Chem. Phys. **106**, 4665 (1997).

[56] B. Space, H. Rabitz, and A. Askar, J. Chem. Phys. **99**, 9070 (1993).

[57] M.E. Tuckerman, B.J. Berne, and A. Rossi, J. Chem. Phys. **94**, 1465 (1991).

[58] G. Zhang and T. Schlick, J. Chem. Phys. **101**, 4995 (1994).

[59] G. G. Batrouni, G. R. Katz, A. S. Kronfeld, G. P. Lepage, B. Svetitsky and K. G. Wilson, Phys. Rev. D **32**, 2736 (1985).

[60] F. J. Alexander, B. M. Boghosian, R. C. Brower, and S. R. Kimura, Phys. Rev. E **64**, 066704 (2001).

[61] B. Dünweg, private discussion, 2009.

[62] D. Goldfarb, Math. Comp. **30**, 796 (1976).

[63] K.W. Brodlie, A. R. Gourlay and J. Greenstadt, J. Inst. Math. Appl. **11**, 73 (1973).

[64] D. C. Liu and J. Nocedal, Math. Program. **45**, 503 (1989).

[65] D.J. Wales and J.P.K. Doye, J. Phys. Chem. A **101**, 5111 (1997).

[66] R. Fletcher and M.J.D. Powell, Computer J. **6**, 163 (1963).

[67] M. T. Chu, R. E. Funderlic, and G. H. Golub, SIAM. J. Matrix Anal. & Appl. **20**, 428 (1998).

[68] R.B. Schnabel, Math. Program. **15**, 247 (1978).

[69] W.C. Davidon, Math. Program. **9**, 1 (1975).

[70] C.R. Rao and S.K. Mitra, *Generalized Inverse of Matrices and its Applications*, Wiley, (New York 1971).

[71] M. B. Reed, Int. J. Comput. Math. **86**, 606 (2009).

[72] J. L. McCauley , Phys. A **382**, 445 (2007).

[73] J. L. McCauley , MPRA Paper No. 2128; http://mpra.ub.uni-muenchen.de/2128/, (2007).

[74] P. Eastman and S. Doniach, Proteins **30**, 215 (1998).

[75] B. Mishra and T. Schlick, J. Chem. Phys. **105**, 299 (1996).

[76] L. N. Mazzoni and L. Casetti, Phys. Rev. Lett. **97**, 218104 (2006).

[77] A. Buckley and A. Lenir, Math. Program. **27**, 155 (1983).

[78] D. F. Shanno and K-H Phua, J. Optimiz. Theory App. **25**, 507 (1978).

[79] V. Tozzini, Accounts of Chem. Res. **43**, 220 (2010).

[80] C.D. Chau, G.J.A. Sevink and J.G.E.M. Fraaije, J. Chem. Phys. **128**, 244110 (2008).

[81] C.D. Chau, G.J.A. Sevink and J.G.E.M. Fraaije, Phys. Rev. E. (2009).

[82] J. Nocedal and S.J. Wright, *Numerical Optimisation*, Springer Series in Operation Research, New York (1999).

[83] C.J. Camacho and D. Thirumalai, Proc. Natl. Acad. Sci. USA **92**, (1995).

[84] M. Dadlez and P.S. Kim, Nat. Struct. Biol. **2**, 674 (1995).

[85] L.A. Mirny, V. Abkevich and E.I. Shakhnovich, Fold. Des. **1**, 221 (1996).

[86] K.A. Dill et al. and H.S. Chan, Protein Sci. **4**, 561 (1995).

[87] J.D. Bryngelson, J.N. Onuchic, N.D. Socci and P.G. Wolynes, Proteins **21**, 167 (1995).

[88] P.G. Wolynes, J.N. Onuchic and D. Thirumalai, Science **267**, 1619 (1995).

[89] H.S. Chan and K.A. Dill, J. Chem. Phys. **100**, 9238 (1994).

[90] D. Thirumalai, Theoretical perspectives on in vitro and in vivo protein folding. In: S. Doniach, Editor, Statistical Mechanics, Protein Structure, and Protein Substrate Interactions, Plenum Press, New York (1994).

[91] T. Veitshans, D. Klimov, and D. Thirumalai, Folding Des. **2**, 1 (1997).

[92] L. Angelani and G. Ruocco, EPL **87**, 18002 (2009).

[93] A. Samiotakis, D. Homouz and M. S. Cheung, J. Chem. Phys. **132**, 175101 (2010).

[94] C.G. Broyden, J. Inst. Math. Appl. **6**, 222 (1970).

[95] S. S. Oren and E. Spedicato, Mathematical Programming **10**, 70 (1976).

[96] R. Fletcher, Computer J. **13** 317 (1970).

[97] D.F. Shanno and K.H. Phua , Math. Programming **14**, 149 (1978).

[98] C.G. Broyden, Math. Comput. **21**, 368 (1967).

[99] H. Yabe, H. J. Martinez, and R. A. Tapia, SIAM J. Optim. **15**, 139 (2004) .

[100] S.K. Kearsley, Acta Cryst. A **45**, 208 (1989).

[101] D.F. Shanno, Computers and Chemical Engineering **7**, 569 (1983).

[102] M.J.D. Powell, *The Convergence of Variable Metric Methods for Nonlinearly Constrained Optimization Calculations*, Nonlinear Programming 3, (O.L. Mangasarian, R.R. Meyer and S.M. Robinson, eds.), Academic Press (1978).

[103] M.J.D. Powell, *A Fast Algorithm for Nonlinearly Constrained Optimization Calculations*, Numerical Analysis, G.A.Watson ed., Lecture Notes in Mathematics, Springer Verlag, Vol. 630, (1978).

[104] D.L. Pincus, S.S. Cho, C. Hyeon and D. Thirumalai, Progress in Molecular Biology and Translational Science **84**, 203 (2008).

[105] P.J. Flory, *Statistical mechanics of chain molecules*, Interscience publishers, New York (1969).

[106] A. Laio, M. Parrinello, Proc. Natl. Acad. Sci. USA. **99**, 12562 (2002).

[107] J.P. Ryckaert, G. Ciccotti G and H.J.C. Berendsen, J. Comput. Phys. **23**, 327 (1977).

[108] H.C. Andersen, J. Comput. Phys. **52**, 24 (1983).
[109] B. Hess, H. Berendsen and J.G.E.M. Fraaije, J. Comput. Chem. **18**, 1463 (1997).
[110] O.M.J. Crippen, J. Camp. Phys. **24**, 96 (1977).
[111] A.L. Mackay, Acta Crystallogr. Sect. A **30**, 440 (1974).

APPENDIX A

Langevin equation with space dependent mobility and its discretised form

A.1. Derivation of the general Langevin equation and numerical implementation

According to Gardiner [9] the many variable version of the Fokker Planck equation, which describes the time evolution of the probability density function of a stochast $\mathbf{x}$ is given as

$$\frac{\partial p(\mathbf{x}, t|\mathbf{x}_0, t_0)}{\partial t} = -\sum_{i=1}^{n} \partial_i (p(\mathbf{x}, t|\mathbf{x}_0, t_0) a_i(\mathbf{x})) + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \partial_i \partial_j (p(\mathbf{x}, t|\mathbf{x}_0, t_0) D_{ij}(\mathbf{x})), \quad \text{(A-1)}$$

which is related to the stochastic differential equation

$$\mathrm{d}\mathbf{x} = \mathbf{a}(\mathbf{x})\mathrm{d}t + B(\mathbf{x})\mathrm{d}\mathbf{W}, \quad \text{(A-2)}$$

by $D(\mathbf{x}) = B(\mathbf{x})B(\mathbf{x})^T$. The drift vector $\mathbf{a}(\mathbf{x})$ and noise matrix $B(\mathbf{x})$ are obtained by requiring the stationary solution of the FPE $p_s(\mathbf{x})$ to be the Boltzmann distribution

$p_s(\mathbf{x}) = N\exp\left[-\beta\phi(\mathbf{x})\right]$. Setting $\frac{\partial p(\mathbf{x},t|\mathbf{x}_0,t_0)}{\partial t} = 0$ and substitute $p_s(\mathbf{x})$ for $p(\mathbf{x},t|\mathbf{x}_0,t_0)$ in the FPE gives

$$\begin{aligned}
p_s(\mathbf{x})a_i(\mathbf{x}) &= \frac{1}{2}\sum_{j=1}^{n}\partial_j\left(p_s(\mathbf{x})D_{ij}(\mathbf{x})\right) && \text{(A-3)}\\
&= \frac{1}{2}\sum_{j=1}^{n}\left(D_{ij}(\mathbf{x})\partial_j p_s(\mathbf{x}) + p_s(\mathbf{x})\partial_j D_{ij}(\mathbf{x})\right) && \text{(A-4)}\\
&= \frac{1}{2}\sum_{j=1}^{n}\left(-\beta D_{ij}(\mathbf{x})\partial_j\phi(\mathbf{x})p_s(\mathbf{x}) + p_s(\mathbf{x})\partial_j D_{ij}(\mathbf{x})\right) && \text{(A-5)}\\
&\rightarrow \\
a_i(\mathbf{x}) &= \frac{1}{2}\sum_{j=1}^{n}\left(-\beta D_{ij}(\mathbf{x})\partial_j\phi(\mathbf{x}) + \partial_j D_{ij}(\mathbf{x})\right). && \text{(A-6)}
\end{aligned}$$

This leads to the following SDE

$$\mathrm{d}\mathbf{x} = \frac{1}{2}\left[-\beta D(\mathbf{x})\nabla\phi(\mathbf{x}) + \nabla D(\mathbf{x})\right]\mathrm{d}t + B(\mathbf{x})\mathrm{d}W(t), \tag{A-7}$$

where $B(\mathbf{x})B^T(\mathbf{x}) = D(\mathbf{x})$ and $\beta^{-1} = k_BT$. After defining $D(\mathbf{x}) = 2k_BTM(\mathbf{x}) = 2k_BTL(\mathbf{x})L^T(\mathbf{x})$ one obtains

$$\mathrm{d}\mathbf{x} = \left[-M(\mathbf{x})\nabla\phi(\mathbf{x}) + k_BT\nabla\cdot M(\mathbf{x})\right]\mathrm{d}t + \sqrt{2k_BT}L(\mathbf{x})\mathrm{d}W(t), \tag{A-8}$$

where the noise term satisfies the fluctuation dissipation theorem. Equation (A-8) is equivalent to the SDE proposed by Hütter and Öttinger [39]

$$\mathrm{d}\mathbf{x} = \left[-M(\mathbf{x})\nabla\Phi(\mathbf{x})\right]\mathrm{d}t + \frac{1}{2}\left[M(\mathbf{x}+\mathrm{d}\mathbf{x})M(\mathbf{x})^{-1} + I\right]\sqrt{2k_BT}L(\mathbf{x})\mathrm{d}W(t). \tag{A-9}$$

This can easily proven by expanding $M(\mathbf{x}+\mathrm{d}\mathbf{x})$ around $\mathbf{x}$ and obeying the rules $\mathrm{d}W\mathrm{d}t = 0$ and $\mathrm{d}W\mathrm{d}W = \mathrm{d}t$.

The discretized form of the SDE proposed by Hütter and Öttinger is given below. The update for $\mathbf{x}_k$ at simulation step k is given as

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{x}_k - \frac{1}{2}\left[M(\mathbf{x}_k + \Delta\mathbf{x}_k^p)\nabla\Phi(\mathbf{x}_k + \Delta\mathbf{x}_k^p) + M(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\right]\Delta t\\
&\quad + \frac{1}{2}\left[M(\mathbf{x}_k + \Delta\mathbf{x}_k^p)M^{-1}(\mathbf{x}_k) + I\right]\sqrt{2k_BT}L(\mathbf{x}_k)\Delta W_t, && \text{(A-10)}
\end{aligned}$$

with the corresponding predictor step

$$\Delta \mathbf{x}_k^p = -M(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\Delta t + \sqrt{2k_BT}L(\mathbf{x}_k)\Delta W_t. \tag{A-11}$$

The approximate inverse Hessian $B(\mathbf{x}_k) = B_k$, which is taken as the mobility tensor $M(\mathbf{x}_k)$, can be obtained using the DFP update

$$B_{k+1} = B_k - \frac{B_k\mathbf{y}_k\mathbf{y}_k^T B_k}{\mathbf{y}_k^T B_k\mathbf{y}_k} + \frac{\mathbf{s}_k\mathbf{s}_k^T}{\mathbf{y}_k^T\mathbf{s}_k}, \tag{A-12}$$

where,

$$\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \quad \text{and} \quad \mathbf{y}_k = \nabla\Phi(\mathbf{x}_{k+1}) - \nabla\Phi(\mathbf{x}_k). \tag{A-13}$$

APPENDIX B

Derivation of the (limited) factorised secant update scheme

B.1. Predictor-corrector scheme for the spurious drift term

The generalized S-QN equation is given by

$$\mathrm{d}\mathbf{x} = [-B(\mathbf{x})\nabla\Phi(\mathbf{x}) + k_B T \nabla \cdot B(\mathbf{x})]\mathrm{d}t + \sqrt{2k_B T}\, J(\mathbf{x})\mathrm{d}W(t). \tag{B-1}$$

We have previously shown [46] that (B-1) can be discretized using the predictor-corrector scheme introduced by Hütter and Öttinger [39] as

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{x}_k + \Delta\mathbf{x}_k, && \text{(B-2)}\\
\Delta\mathbf{x}_k &= -\frac{1}{2}[B(\mathbf{x}_k + \Delta\mathbf{x}_k^p)\nabla\Phi(\mathbf{x}_k + \Delta\mathbf{x}_k^p) + B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)]\Delta t \\
&+ \frac{1}{2}[B(\mathbf{x}_k + \Delta\mathbf{x}_k^p)B^{-1}(\mathbf{x}_k) + I]\sqrt{2k_B T}\, J(\mathbf{x}_k)\Delta W_t, && \text{(B-3)}\\
\Delta\mathbf{x}_k^p &= -B(\mathbf{x}_k)\nabla\Phi(\mathbf{x}_k)\Delta t + \sqrt{2k_B T}\, J(\mathbf{x}_k)\Delta W_t. && \text{(B-4)}
\end{aligned}$$

where (B-4) is the predictor step and (B-2) is the correction. Direct inversion of B would be costly and should therefore be avoided. Using the Sherman-Morrison theorem, the exact inverse $G_k = G(\mathbf{x}_k) = B^{-1}(\mathbf{x}_k)$ of $B(\mathbf{x}_k)$ in dual space can be calculated explicitly by

$$G_k = (I - \frac{\mathbf{y}_{k-1}\mathbf{s}_{k-1}^T}{\mathbf{y}_{k-1}^T\mathbf{s}_{k-1}})G_{k-1}(I - \frac{\mathbf{s}_{k-1}\mathbf{y}_{k-1}^T}{\mathbf{y}_{k-1}^T\mathbf{s}_{k-1}}) + \frac{\mathbf{y}_{k-1}\mathbf{y}_{k-1}^T}{\mathbf{y}_{k-1}^T\mathbf{s}_{k-1}}, \tag{B-5}$$

reusing the vectors $\mathbf{y}_{k-1}$ and $\mathbf{s}_{k-1}$ stored for updating B_{k-1}. Disregarding the costs associated with the computation of $\nabla\Phi(\mathbf{x}_k + \Delta\mathbf{x}_k^p)$ and the storage of G, we can calculate the costs of this predictor-corrector scheme employed for general Φ. For quadratic potentials, when the predictor (B-4) suffices and $\Delta\mathbf{x}_k = \Delta\mathbf{x}_k^p$, the total costs are $7n^2$ (see the theory section in Chapter 3). Due to the very related structure, the corrector equation (B-2) is $7n^2$ as well (if we reuse terms) plus an additional $2n^2$ for $B^{-1}(\mathbf{x}_k)$ using (B-5). The additional costs for (B-2) are thus $9n^2$ and the total costs for the predictor-corrector scheme using FSU are $16n^2$. The Sherman-Morrison theorem can also be applied to derive an analytic expression for $D_k = J_k^{-1}$ from (B-13), providing an efficient method for determining $B^{-1}(\mathbf{x}_k)$ for L-FSU. Again, the total costs of the full scheme are roughly doubled compared to using only the predictor term. Since this calculation is straightforward but involved, the full technical details are given in future publications for general Φ. As a concluding remark, we note that the calculation of the divergence itself may actually be more efficient than the predictor-corrector scheme, because of the special nature of the update $B(\mathbf{x}_k) = B(\mathbf{x}_{k-1}) + V$, with V a rank-two correction.

B.2. Derivation of the FSU algorithm

The derivation of the update for J is equivalent to the update for the lower triangular matrix L [33]. By interchanging $\mathbf{s}$ and $\mathbf{y}$ and replacing L with J, the matrices LL^T and JJ^T become approximates of the Hessian and the inverse Hessian respectively. Here we focus on the derivation of the update scheme for J.
Given $\|\cdot\|$ is the Frobenius norm and

$$\min_{J_{k+1}} \|J_{k+1} - J_k\|, \tag{B-6}$$

$$J_{k+1}\mathbf{v}_k = \mathbf{s}_k, \tag{B-7}$$

J_{k+1} is uniquely given by

$$J_{k+1} = J_k + \frac{\mathbf{s}_k - J_k\mathbf{v}_k}{\mathbf{v}_k^T\mathbf{v}_k}\mathbf{v}_k^T. \tag{B-8}$$

Substitute J_{k+1} into

$$J_{k+1}^T \mathbf{y}_k = \mathbf{v}_k, \tag{B-9}$$

gives

$$\begin{aligned}\mathbf{v}_k = J_{k+1}^T \mathbf{y}_k &= \left(J_k + \frac{\mathbf{s}_k - J_k \mathbf{v}_k}{\mathbf{v}_k^T \mathbf{v}_k} \mathbf{v}_k^T \right)^T \mathbf{y}_k, \\ &= J_k^T \mathbf{y}_k + \frac{(\mathbf{s}_k - J_k \mathbf{v}_k)^T \mathbf{y}_k}{\mathbf{v}_k^T \mathbf{v}_k} \mathbf{v}_k \qquad \text{(B-10)} \\ &\Rightarrow \\ (1 - \frac{(\mathbf{s}_k - J_k \mathbf{v}_k)^T \mathbf{y}_k}{\mathbf{v}_k^T \mathbf{v}_k}) \mathbf{v}_k &= J_k^T \mathbf{y}_k. \qquad \text{(B-11)}\end{aligned}$$

Hence, $\mathbf{v}_k = \alpha_k J_k^T \mathbf{y}_k$ and after substituting this into (B-10) gives

$$\alpha^2 = \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{y}_k^T J_k J_k^T \mathbf{y}_k}, \tag{B-12}$$

which has a real solution for α due to the curvature condition and positive definiteness of $J_k J_k^T$. The update scheme for J_{k+1} is now given by

$$J_{k+1} = J_k + \frac{\alpha \mathbf{s}_k \mathbf{y}_k^T J_k - \alpha^2 J_k J_k^T \mathbf{y}_k \mathbf{y}_k^T J_k}{\mathbf{y}_k^T \mathbf{s}_k}. \tag{B-13}$$

Using this update we find after some algebraic operations that JJ^T is equal to the update derived from the BFGS scheme

$$\begin{aligned}J_{k+1} J_{k+1}^T &= J_k J_k^T - \frac{J_k J_k^T \mathbf{y}_k \mathbf{y}_k^T J_k J_k^T}{\mathbf{y}_k^T J_k J_k^T \mathbf{y}_k} + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}, \\ &= B_k - \frac{B_k \mathbf{y}_k \mathbf{y}_k^T B_k}{\mathbf{y}_k^T B_k \mathbf{y}_k} + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}. \qquad \text{(B-14)}\end{aligned}$$

B.3. The limited memory update scheme

We consider our L-FSU method in the framework of limited-memory approaches. To arrive at a limited-memory BFGS method, two different strategies have been used.

The L-BFGS method of Liu and Nocedal [64] recasts BFGS into a multiplicative form $B_{k+1} = V_k^T B_k V_k + \rho_k s_k s_k^T$, and truncates by only using the information stored in V_k and s_k during the last m updates. In particular, given a (often diagonal) B_0, the L-BFGS update is provided by

$$\begin{aligned} B_{k+1} &= (V_k^T \ldots V_{k-m+1}^T) B_0 (V_{k-m+1} \ldots V_k) \\ &\quad + \rho_{k-m+1} (V_k^T \ldots V_{k-m+2}^T) s_{k-m+1} s_{k-m+1}^T (V_{k-m+2} \ldots V_k) \\ &\quad + \rho_{k-m+2} (V_k^T \ldots V_{k-m+3}^T) s_{k-m+2} s_{k-m+2}^T (V_{k-m+3} \ldots V_k) \\ &\quad + \rho_{k-1} V_k^T s_{k-1} s_{k-1}^T V_k + \rho_k s_k s_k^T. \end{aligned} \tag{B-15}$$

This approach was recently generalized by Reed [71] for the convex Broyden family of Quasi-Newton updates. The variable storage conjugate gradient (VSCG) method of Buckley and LeNir [77] is based on the BFGS formula in the additive form and overwrites the most recent update once m is reached. If only the current update is stored, both algorithms reduce to the memoryless QN method of Shannon and Phua [78]. It is generally recognized that L-BFGS with Shanno scaling is the most efficient and reliable method across a range of test problems.

We can rewrite the update scheme for J_{k+1} in (B-13) as $J_{k+1} = V_k J_k = (\prod_{j=0}^{k} V_{k-j}) J_0$ with

$$V_k = (I - \frac{1}{\nu_k} \upsilon_k y_k^T), \tag{B-16}$$

with $\upsilon_k = h_k - s_k/\alpha_k$, $h_k = J_k J_k^T y_k$ and $\nu_k = h_k^T y_k$. Using the additional condition, $B_{k+1} = J_{k+1} J_{k+1}^T$, we obtain

$$\begin{aligned} B_{k+1} = J_{k+1} J_{k+1}^T &= V_k V_{k-1} \ \ldots \ V_0 J_0 J_0^T V_0^T \ \ldots \ V_{k-1}^T V_k^T && \text{(B-17)} \\ &= V_k J_k J_k^T V_k^T = V_k B_k V_k^T. && \text{(B-18)} \end{aligned}$$

Rewriting this expression in the additive form, several terms cancel and we obtain exactly the additive Davidon-Fletcher-Powell (DFP) formula (see also appendix B.2) [71]. Hence, the multiplicative DFP formula

$$B_{k+1} = V_k^T B_k V_k + \rho_k s_k s_k^T \quad \text{with} \quad \mathrm{V_k = (I - \frac{1}{\nu_k} y_k h_k^T)}, \tag{B-19}$$

and the update scheme in FSU are equivalent. The principle difference is that we casted (B-19) into a *factorized* form (B-18). The recursive expression (B-17), obtained by loop unrolling, can serve as a basis for limited-memory implementation.

The recursive algorithm also allows for a limitation of the memory requirements of FSU, by storing at each k step vectors $\{y_k, s_k, h_k\}$ instead of matrices J_k and B_k in the original scheme (B-13), however, at the expense of an additional computational load. Since (B-13) is multiplicative, we adapt the L-BFGS strategy for limited-memory implementation of FSU (L-FSU). However, instead of truncating the incorporation of V_k in B, we truncate in J, i.e.

$$J_{k+1} = V_k V_{k-1} ... V_{k-m+1} J_0, \tag{B-20}$$

for $k \geq m$, and apply the second relation to update the mobility B

$$J_{k+1} J_{k+1}^T \quad = \quad V_k V_{k-1} \quad ... \quad V_{k-m+1} J_0 J_0^T V_{k-m+1}^T \quad ... \quad V_{k-1}^T V_k^T . \tag{B-21}$$

For $k < m$, the FSU relations apply. Upon comparing L-FSU to L-BFGS in (B-15), with V_k as in (B-19), we note three important properties: a) L-FSU is factorized, b) the memory requirements of L-FSU are the same as in L-BFGS, c) assuming $B_0 = I$, the number of matrix-vector products in L-FSU ($2m$) is of a different order than L-BFGS ($2m + m(m-1)$, or $2m + m(m-1)/2$ if case of re-using information). One remaining issue is whether the secant condition is satisfied by L-FSU for $k \geq m$.The L-Broyden family [71] was specially designed to satisfy the secant condition $B_{k+1} y_k = s_k$ for all k, since $V_k y_k = 0$. By construction, the L-FSU method satisfies the secant condition for $k < m$. Let $m > 1$ and $k \geq m$, we define a matrix $\tilde{B}_k = \tilde{J}_k \tilde{J}_k^T$ by

$$\tilde{J}_k = V_{k-1} ... V_{k-m+1} J_0, \tag{B-22}$$

and we find that

$$B_{k+1} y_k = J_{k+1} J_{k+1}^T y_k = \alpha_k (\tilde{h}_k - \beta_k h_k) + \beta_k s_k \ . \tag{B-23}$$

with $\tilde{h}_k = \tilde{B}_k y_k$ and $\beta_k = \tilde{h}_k^T y_k / h_k^T y_k$. Consequently, the secant condition is satisfied only when $\tilde{h}_k = h_k = J_k J_k^T y_k$, which is generally not the case. We now redefine V_k as

$$V_k = (I - \frac{1}{\tilde{h}_k^T y_k} (\tilde{h}_k - \frac{s_k}{\tilde{\alpha}_k}) y_k^T), \tag{B-24}$$

with $\tilde{\alpha}_k^2 = s_k^T y_k / \tilde{h}_k^T y_k$. Substituting this into (B-21) gives

$$J_{k+1} J_{k+1}^T \mathbf{y}_k = V_k \tilde{B}_k V_k^T \mathbf{y}_k = \tilde{\alpha}_k V_k \tilde{B}_k \mathbf{y}_k = \tilde{\alpha}_k V_k \tilde{\mathbf{h}}_k = \mathbf{s}_k, \tag{B-25}$$

and the secant condition is again satisfied. We note that only the h_k for $k \geq m$ are affected by this redefinition of V_k.

B.4. Recursive scheme for the limited memory update

The update scheme can be casted into
Algorithm 1.

```
$\mathbf{d} = \mathbf{d}(\mathbf{x}_{K+1});$                                  (B-26)

for $i = K, ..., \max(0, K - m + 1)$
    $\mathbf{v}_i = \mathbf{h}_i - \mathbf{s}_i/\alpha_i;$
    $\lambda_i = \mathbf{v}_i^T \mathbf{d};$                                   (B-27)
    $\mathbf{d} = \mathbf{d} - (\lambda_i/\mathbf{h}_i^T \mathbf{y}_i)\mathbf{y}_i;$
end

$\mathbf{d} = J_0 J_0^T \mathbf{d};$                                           (B-28)

for $i = \max(0, K - m + 1), ..., K$
    $\gamma_i = \mathbf{y}_i^T \mathbf{d};$
    $\beta_i = \mathbf{y}_i^T \mathbf{d};$                                     (B-29)
    $\mathbf{d} = \mathbf{d} - (\beta_i/\mathbf{h}_i^T \mathbf{y}_i)\mathbf{v}_i;$
end

stop with result $\mathbf{d} = J(\mathbf{x}_{K+1})J(\mathbf{x}_{K+1})^T \mathbf{d}.$   (B-30)
```

It is clear that for $K = k$, the procedure in Algorithm 1 provides the drift term in (3.6) for $\mathbf{d} = \mathbf{d}(\mathbf{x}_{k+1}) = -\nabla\Phi(\mathbf{x}_{k+1})\Delta t$ in (B-26). The noise term can be calculated using the second part of Algorithm 1, starting with (B-28) and $\mathbf{d} = \sqrt{2k_B T} J_0 \Delta W_t$. For $k < m$, the vector $\mathbf{h}_k = J_k J_k^T \mathbf{y}_k$ can also be obtained using Algorithm 1 by setting $\mathbf{d} = \mathbf{y}_k$ and $K = k - 1$. Consequently, we obtain α_k from

$$\alpha_k = \alpha_k(\mathbf{h}_k) = \sqrt{\frac{\mathbf{s}_k^T \mathbf{y}_k}{\mathbf{h}_k^T \mathbf{y}_k}}, \tag{B-31}$$

and store this new value α_k in a vector α. For $k \geq m$, $\tilde{\mathbf{h}}_k = \tilde{B}_k \mathbf{y}_k$ can be obtained from Algorithm 1 starting with $\mathbf{d} = \mathbf{y}_k$ with the recursive index running between $k - 1$

to $k - m + 1$. We store $\alpha_k = \alpha_k(\tilde{\mathbf{h}}_k)$ and $\mathbf{h}_k = \tilde{\mathbf{h}}_k = \mathbf{d}$. This scheme requires only permanent storage of vector-triplets $\{\mathbf{s}_k, \mathbf{y}_k, \mathbf{h}_k\}$ (each of length n) for each iteration step k. In agreement with general practice the small additional effort for storing and calculating the vector α of length m is not considered in the analysis [21].

Upon analysing the computational load, operations (B-27) and (B-29) add up to $3mn$ and $2mn$ multiplications, respectively. An additional n operations are needed for (B-28), if we assume J_0 is a diagonal (positive definite) matrix, giving rise to $5mn + n$ operations. Recursive calculation of $\mathbf{h}_k$ requires a maximum of $5mn + n$ operations (for $k = m - 1$), and slightly less for other k. The total is a maximum of $10mn + 2n$ multiplications per step for the drift term only. For the noise term only the second part of the algorithm is required. Assuming again a diagonal J_0, we find that n multiplications are required for $\sqrt{2k_BT}J_0\Delta W_t$ and $2mn$ multiplications for (B-29). This brings us to a total of $2mn + n$ multiplications for the noise term, and a total of $12nm + 3n$ for the complete cycle at time step k.

Summary

In this thesis I consider the Langevin equation with an alternative mobility matrix. One often considers the mobility matrix in the Langevin equation as a constant matrix, whereas here I consider the mobility matrix as a space-dependent matrix which needs to be updated at every time step. The Langevin equation is a stochastic differential equation which describes the time evolution of a set of changing macroscopic variables. This set of variables changes relatively slow compared to other (microscopic) variables. One of the earliest descriptions with the Langevin equation is the description of Brownian dynamics: the motion of (macroscopic) pollen grains in water due to the rapidly moving water molecules.

In a molecular system, the positions (coordinates) of the larger-sized particles (molecules/atoms) can be considered as a set of macroscopic variables and the positions of the smaller-sized particles as microscopic variables. This relates to the relatively fast movements of the smaller particles with respect to the movements of the larger particles. The subdivision of the variables enables the use of the Langevin equation for describing these coarse-grained models.

In a simulation with conventional molecular dynamics (MD), the atoms/molecules are represented as particles and the movements of these particles are calculated from Newton's equation of motion. However, simulations of a complex system with many different length and time scales based on the fundamental equations of motion take a very long simulation time before capturing the functional and relevant motions. This problem is called critical slowing down. To avoid this problem multi-scale simulation methods are applied, which permits the use of different size time steps and thus enables acceleration of the relevant (slow) movements.

The aim of this thesis is to develop a stochastic quasi Newton (S-QN) method, such that by incorporating multi-scaling the relevant motions are effectively taken with a

larger time step. Due to the integration of the slow motions with a larger time step, critical slowing down can be avoided. At every time step the slowest modes (corresponding to the relevant movements) can be detected and the acceleration of the movements towards these modes is a possibility. However, detecting the different modes of a system at each time step is computationally very expensive. Therefore, a method that automatically changes the mobility matrix in the Langevin equation is developed, such that in the resulting movements the slow modes are automatically integrated with a larger time step.

In Chapter 2, the new mobility matrix is presented: the inverse Hessian of the energy potential. Due to this choice for the mobility matrix, the deterministic term of the Langevin equation is similar to the displacement in the Newton minimization method. We expect, analogous to the performance difference between the Newton method and the steepest descent method, a better convergence towards the minimum of the energy potential compared to a constant mobility matrix. The stochastic term (noise term) in the Langevin equation, constructed using the factorized mobility, is needed for having correct thermodynamic sampling. Simple one and two dimensional systems show that using the alternative mobility matrix is thermodynamically consistent: the potential energy has a Boltzmann distribution. The choice for the inverse Hessian contributes to a faster barrier crossing. This has been determined quantitatively using the mean first passage time, which measures the average time needed to cross a barrier (getting to the top of the energy landscape starting from the bottom). Using the alternative mobility matrix, the mean first passage time is one order of magnitude smaller than for the constant mobility matrix. Since the calculation of the inverse Hessian is computationally expensive, an approximation for the inverse Hessian has been used. This approximation is derived from an existing update scheme in the quasi Newton method: the DFP-update scheme.

In Chapter 3, a factorized secant update (FSU) is constructed. This update is a factorized equivalent of the DFP-update, such that no explicit factorisation of the mobility is needed, which reduces the complexity from $O(n^3)$ to $O(n^2)$. The FSU is constructed such that it forms a basis for the construction of a limited memory version (LFSU), where no matrices are stored and no matrix multiplications are performed. Instead of storing n^2 matrix elements, $3m$ vectors of length n are stored, where m is the truncation parameter or the history depth. Applying the new update scheme to a harmonic chain of particles shows that the update scheme converges to the inverse Hessian. The FSU and LFSU contribute automatically to the acceleration of the different length and time scales, which results in a better sampling performance compared to the conventional Langevin dynamics (constant mobility matrix).

In Chapter 4, the Langevin dynamics, with the mobility matrix constructed by FSU, is applied to a model protein. For the stability of the simulation, the FSU-scheme has been adapted such that regularization occurs for ill-conditioned systems; i.e. systems with large condition numbers (largest eigenvalue of the mobility matrix divided by the smallest eigenvalue). The model protein is based on 3 types of coarse grained particles which either represent a hydrophobic or hydrophilic or a neutral peptide. This protein model enables the determination of sequence-dependent properties of the protein. Using the S-QN method the slow modes are accelerated, which enables faster crossing of energy barriers and samples a larger area of the energy landscape. As a consequence, different inherent states of the protein are found, including the native state. Detailed analysis shows that automatic multi-scaling first minimizes the bond length, bond angle and torsion potentials to its equilibrium value. Afterwards, a sudden drop in the total potential occurs caused by the non-bonded interaction which starts to play a role. The sudden drop in the total potential coincides with a drop in the overlap function. The overlap function is a measure of the amount of (conformation-wise) overlap with a reference structure, for which the native structure is often taken, such that the overlap function is zero for total overlap and one for no overlap at all. The evolution of the overlap function of the partial structures (sub-domains) indicates that these domains are already established before the drop in the total overlap occurs and that the partial structures move collectively. In the conventional Langevin dynamics the collective movements are absent and less inherent states are found due to critical slowing down.

The proposed FSU scheme enables automatic multi-scaling in the Langevin dynamics and contributes to efficient calculation of the noise term. The construction of the FSU also enables the construction of a limited memory version for the mobility. However, the Langevin equation with a space dependent mobility has a spurious drift term, so that calculation of the divergence of the mobility is needed. This term is absent in the conventional Langevin equation (divergence of a constant mobility has no contribution). The computationally expensive calculation of the divergence of the mobility is avoided by using a prediction and correction scheme, which requires the calculation of the inverse mobility. In Chapter 5, the computation of the Langevin equation with space dependent mobility is considered. For the inverse of the mobility a limited memory version is constructed, so that the whole Langevin equation can be written in a limited memory variant. Standard computational methods are of $O(n^3)$, FSU is of $O(n^2)$ and LFSU is of $O(mn)$ and requires less storage. Together with the multi-scaling feature of (L)FSU, a powerful method for molecular simulations has been provided.

Samenvatting

In dit proefschrift heb ik de Langevin vergelijking beschouwd met een alternatieve mobiliteits matrix. Waar men veelal de mobiliteits matrix in de Langevin vergelijking als constant betracht (inverse van de frictie coëfficiënt), beschouw ik de mobiliteits matrix als een plaatsafhankelijke matrix die in elke simulatie stap vernieuwd moet worden. De Langevin vergelijking is een stochastische differentiaalvergelijking die een beschrijving geeft van de tijdsevolutie van een set veranderende macroscopische variabelen. Deze set van variabelen veranderen relatief langzaam ten op zichte van andere (microscopische) variabelen. Een van de eerste beschrijvingen die gebruik maakt van de Langevin vergelijking is de beschrijving van Brownse beweging: de beweging van (macroscopisch grote) pollen in water door botsingen met de snel bewegende water moleculen.

In een moleculair systeem kunnen de posities (coördinaten) van de grote deeltjes (moleculen/atomen) gezien worden als een set macroscopische variabelen en de posities van de kleine deeltjes kunnen gezien worden als de microscopische variabelen. Dit komt overeen met de relatief snellere bewegingen van de kleinere deeltjes ten opzichte van de bewegingen van de grotere deeltjes. De onderverdeling in variabelen maakt de Langevin vergelijking uiterst geschikt voor de beschrijving van deze "coarse-grained"modellen.

In een simulatie, gebruikmakend van de klassieke moleculaire dynamica (MD), worden de atomen voorgesteld als deeltjes en de bewegingen van deze deeltjes volgen uit de bewegingsvergelijkingen van Newton. Een complex systeem heeft echter vele verschillende lengte- en tijdschalen waardoor simulaties gebaseerd op fundamentele bewegingsvergelijkingen heel veel reken/simulatie-tijd vergen voor het in kaart brengen van functionele en relevante bewegingen. Dit probleem wordt "critical slowing down"genoemd. Om dit probleem te overbruggen worden (behalve coarse-

grained modellen) multi-schaal simulatie technieken toegepast, die verschillende tijdstap groottes toestaan en dus mogelijk maken om de relevante (langzame) bewegingen te accelereren.
Het doel van dit proefschrift is een stochastische quasi Newton (S-QN) methode te ontwikkelen, waarbij door multi-schaling de relevante bewegingen effectief met een grotere tijdstap worden genomen. Door langzame bewegingen met grotere tijdstappen te integreren wordt "critical slowing down"voorkomen. Het detecteren van de langzaamste modes (deze corresponderen met de relevante bewegingen) in elke tijdstap en de bewegingen in deze modes accelereren is een mogelijkheid. Echter, het detecteren van de verschillende modes in een systeem in elke tijdstap is veel te kostbaar. Vandaar de ontwikkeling van een methode die automatisch de mobiliteits matrix in de Langevin vergelijking aanpast, zodanig dat in de resulterende bewegingen de langzame modes automatisch met een grotere tijdstap worden geïntegreerd.
In hoofdstuk 2 wordt de nieuwe mobiliteits matrix gepresenteerd: de inverse Hessiaan van de energie potentiaal. Door deze keuze voor de mobiliteits matrix te nemen lijkt de deterministische term van de Langevin vergelijking op de verplaatsing in de Newton minimalisatie methode. We verwachten, analoog aan het prestatieverschil tussen de Newton methode en de steepest descent methode, een betere convergentie naar het energie minimum vergeleken met een constante mobiliteits matrix. De stochastische term (ruis-term) in de Langevin vergelijking, geconstrueerd met de gefactoriseerde mobiliteit, is essentieel voor correcte thermodynamische distributies. Aan de hand van eenvoudige 1 en 2 dimensionale systemen wordt aangetoond dat de alternatieve mobiliteit thermodynamisch consistent is: de energie potentiaal heeft een Boltzmann verdeling. De keuze van de inverse Hessiaan draagt ook bij aan de snellere verplaatsing over energie barrières. Deze laatste is kwantitatief gemeten aan de hand van de gemiddelde eerste passage tijd waarbij gekeken wordt naar de gemiddelde benodige tijd om van onder aan het energielandschap tot aan de top van het energielandschap te komen. Gebruikmakend van de inverse Hessiaan is deze gemiddelde tijd een orde kleiner dan de gemiddelde tijd nodig met een constante mobiliteit. Omdat het uitrekenen van de analytische inverse Hessiaan erg veel rekentijd vergt, wordt de inverse Hessiaan benaderd. Deze benadering is afgeleid van een bestaand update schema in de quasi Newton methode: het DFP-update schema.
In hoofdstuk 3 wordt een gefactoriseerd secant update (FSU) geconstrueerd. Deze update is een gefactoriseerde equivalent van de DFP-update, waardoor er geen expliciete factorisatie van de mobiliteit meer nodig is, hetgeen de complexiteit reduceert van $O(n^3)$ naar $O(n^2)$. De FSU is ook zodanig geconstrueerd dat het een basis vormt

voor de constructie van een gelimiteerd geheugen versie (LFSU), waarbij geen matrices worden opgeslagen en geen matrix vermenigvuldigingen plaatsvinden. In plaats van n^2 matrix elementen opslaan worden er $3m$ vectoren van lengte n opgeslagen, waarbij m de afkap parameter of de historie diepte is. Voor LFSU wordt de complexiteit nog verder gereduceerd naar $O(mn)$. Aan de hand van een systeem met deeltjes waarbij harmonische interactie tussen de buren plaatsvindt, wordt getoond dat het nieuwe update schema naar de inverse Hessiaan convergeert. Ook geldt dat FSU en LFSU automatisch bijdragen aan het accelereren van de verschillende tijd en lengte schalen, met als resultaat een betere sampling prestatie vergeleken met de conventionele Langevin dynamica (constante mobiliteits matrix).

In hoofdstuk 4 wordt de Langevin dynamica met de door FSU geconstrueerde mobiliteit toegepast op een model eiwit. Voor de stabiliteit van de simulatie wordt het FSU schema aangepast zodanig dat er regularisatie plaatsvindt voor slecht geconditioneerde systemen: systemen met grote conditie getallen (grootste eigenwaarde van de mobiliteit matrix gedeeld door de kleinste eigenwaarde). Het model eiwit is gebaseerd op 3 soorten ‘groffe’ deeltjes die ofwel een hydrofobe ofwel een hydrofiele ofwel een neutrale peptide representeren. Met dit eiwit model is het mogelijk om ketenafhankelijke eigenschappen van een eiwit te bepalen. Met de S-QN methode worden de langzame modes geaccelereerd, hetgeen de verplaatsing over energie barrières versneld en een groter gebied van het energie landschap wordt bewandeld. Het gevolg is dat verschillende lokale minima zijn gevonden, waaronder ook het globale minimum. Gedetailleerde analyse van de SQ-N methode laat zien dat automatische multi-schaling bijdraagt aan het eerst equilibreren van de bindings lengtes, bindings hoeken en torsie hoeken. Nadat deze bindings eigenschappen zijn gestabiliseerd wordt er een plotselinge daling in het totale energie potentiaal waargenomen, hetgeen correspondeert met de niet bindings-interacties die een rol gaan spelen. De plotselinge daling in de totale energie potentiaal komt overeen met de daling in de overlap functie. Deze functie geeft aan in hoeverre de configuratie van het eiwit overeenkomt met de configuratie van een referentie eiwit. Als referentie eiwit wordt vaak de natieve conformatie genomen, waarbij een totale overlap correspondeert met functie waarde 0 en geen overlap met functie waarde 1. De evolutie van de overlap functies van de partiële structuren(sub-domeinen) laat zien dat deze domeinen alvorens de plotseling daling al zijn gevormd en collectief bewegen. De collectieve bewegingen zijn afwezig in de conventionele Langevin dynamica zodat minder minima worden gevonden vanwege de kritieke vertraging.

Het voorgestelde FSU schema draagt bij aan de automatische multi-schaling in de

Langevin dynamica en een efficiënte berekening van de ruis-term. Ook is de FSU zodanig opgezet dat een gelimiteerde variant voor het berekenen van de mobiliteit mogelijk is. Echter de Langevin vergelijking met een plaatsafhankelijke mobiliteit bezit een correctie term voor de ruis, waarbij de divergentie van de mobiliteit berekend moet worden. Deze term heeft geen bijdrage in de conventionele Langevin vergelijking (divergentie van constante mobiliteit levert geen bijdrage). Omdat de divergentie term uitrekenen computationeel erg duur is, wordt dit vermeden door een predictie en correctie schema, waarbij wel de inverse van de mobiliteit moet worden uitgerekend. In hoofdtuk 5 wordt op de computationele aspecten van het uitrekenen van de Langevin vergelijking met plaatsafhankelijke mobiliteit ingegaan. Voor de inverse van de mobiliteit wordt een gelimiteerde geheugen schema geconstrueerd, waardoor de gehele Langevin vergelijking in een gelimiteerde geheugen vorm geschreven kan worden. Standaard computationele methoden zijn van $O(n^3)$, terwijl FSU van $O(n^2)$ is en LFSU van $O(mn)$ en minder opslagcapaciteit nodig heeft. Tezamen met de automatische multi-schaling eigenschap, vormt (L)FSU een krachtige methode voor het uitvoeren van moleculaire simulaties.

Curriculum Vitae

The author was born on 25th of November, 1980 in Vlaardingen, the Netherlands. At the age of four he came back from Hong Kong and continued at a primary school in Vlaardingen. After finishing high school at *Aquamarijn College*, Vlaardingen, in 1999 he entered the Technical University of Eindhoven (TUE) for a double majored first year university program; computer science and applied mathematics. Although there was interest towards different exact sciences, he chose to continue with mathematics. During his master study he went to the department of Aeronautical and Vehicle engineering at The Royal Institute of Technology (Kungliga Tekniska Högskolan) in Sweden to work as a trainee on researching the liner of a turbo jet engine. Thereafter his curiosity and interest in science grew. Back in the Netherlands he started to work on a phenomenon in dynamics of vortices observed by the Turbulence and Vortex dynamics group (from the TUE) under supervision of prof.dr. J. Molenaar, which resulted in his master thesis, entitled *2-D flow evolution in bounded domains*. In 2005 he received a master degree in applied mathematics.

In 2005 he started his Ph.D. research with Prof. dr. Hans Fraaije in the Soft Matter Chemistry group at the Leiden University. During his Ph.D. research the author attended the Ph.D. school in Han-sur-Lesse and also several Ph.D. workshops has been attended (courses include: "Polymer Physics", from the national research school PTN (Polymeer Technologie Nederland); "Understanding molecular simulations", from CECAM). The author also participated in workshops outside the chemistry field, including: "Mathematics for Industry" study group and the workshop: "Fast Numerical Solution of Partial Differential Equations" in Utrecht. Poster presentations have been given by the author at the "Vloeistoffen en grensvlakken" meetings in Lunteren, the Netherlands. His Ph.D. work has also been presented orally in his research group and at the meetings in "Vloeistoffen en grensvlakken" in Lunteren.

Acknowledgments

In the end I would like to express my gratitude to people, who in one way or another influenced my life and the realization of this thesis.

First I would like to thank my office mates from the Soft Matter Chemistry and Colloid and Interface Science for creating a good working atmosphere. My special thanks to Hana Robson, Valeriya Ratushna , Frank Versluis and Ahmed el Jilali for having sense and nonsense talk about life. Where with the latter two the (non)sense discussions often continued after office hours during our walk from the office to the train station. During my PhD I was also very glad to meet Edgar Blokhuis. A person hard to forget since he triggered my (educated) gambling addiction, thanks Edgar!

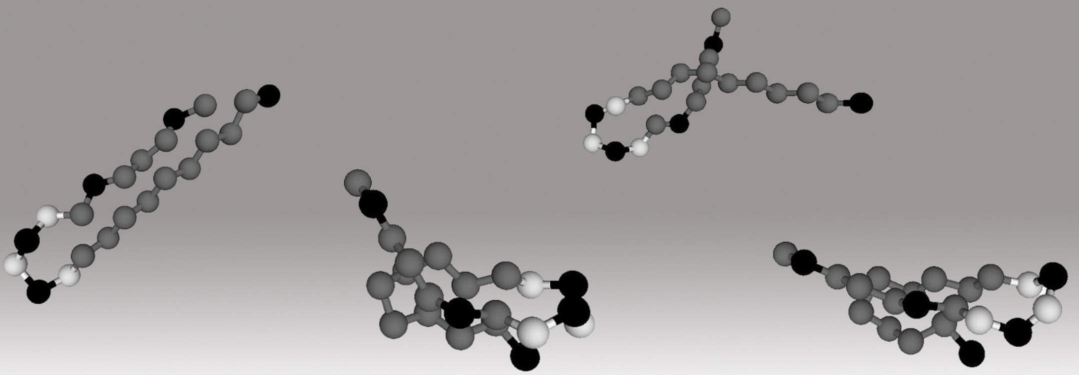
I would like to express my gratitude to the rector magnificus of the Technical University of Eindhoven, Hans van Duijn, who sent me away from TUE to experience how others handle/solve problems (until today I am not sure whether he doesn't want me in Eindhoven or it was a sincere advice). Which brings me to Hans Fraaije, who gave a mathematician the opportunity to work in the physical chemistry field. I am very grateful for this opportunity and also the physical insight that I have gotten from him. Although the physics can be explained, really understanding and being conscious about physical phenomena takes time. One might want to compare the process of realization of this thesis as one whole minimization process, where finding the global minimum is the goal. This thesis provides a acceleration technique for accelerating the slow modes and avoiding critical slowing down. Unfortunately this technique is not present in real life and in the process of thesis realization. However when I got stuck (staying in a local minima) or moving in the wrong direction, there was always Agur Sevink (the accelerator?) who provided me valuable advice and prevented me from getting off road (critical slowing down). Thanks for your help and being present!

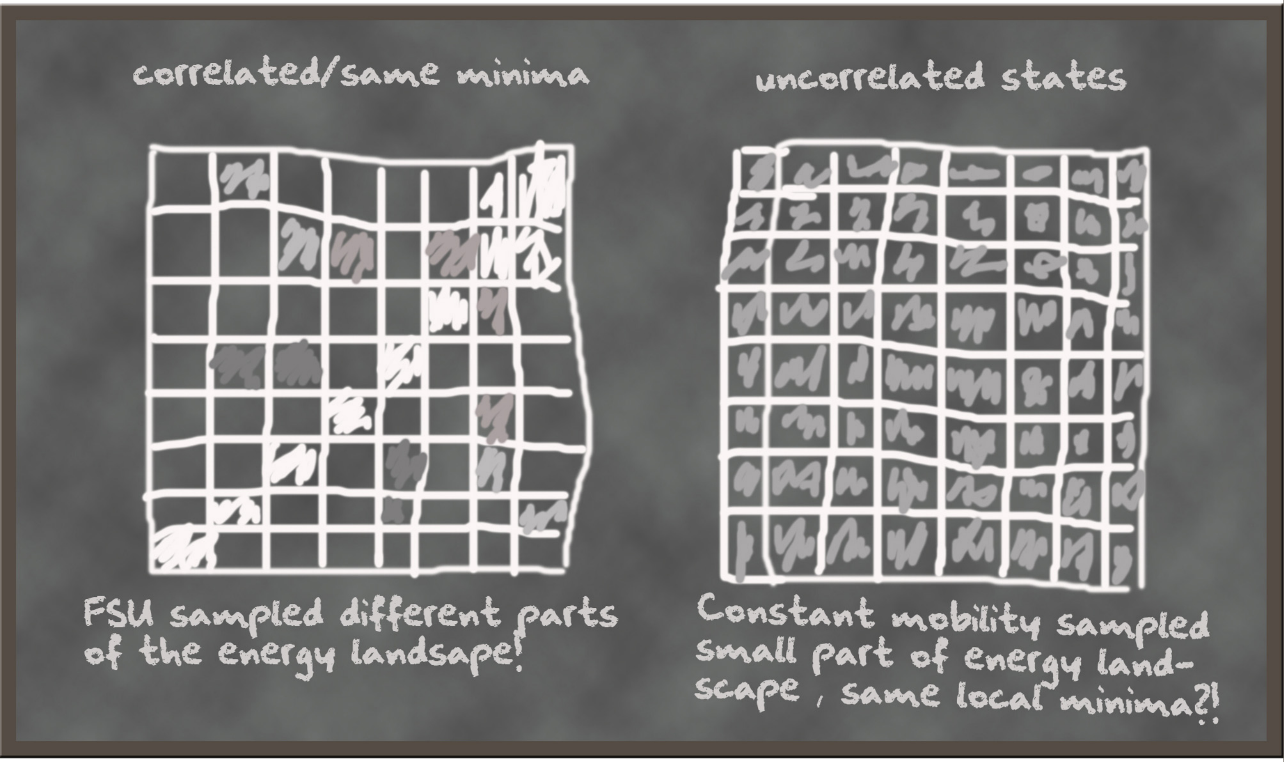
My life would not have been the same without Camron and Olivia Chau. I will never forget our wonderful and meaningful discussions day and night. It was very interesting to experience how you approach and solve problems, which certainly will be cherished in my memory.

Last but not least, my sincere gratitude towards Cherryl Chang and my family for their endless patience and for just being there.

A Stochastic Quasi Newton Method for molecular simulations

Chun Dong Chau





FSU sampled different parts of the energy landsape!

Constant mobility sampled small part of energy landscape, same local minima?!